



วงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

โดย

นายวีรพงษ์ ห้าวหาญ B4608738

นายอัฐพล เรืองบุตร B4611813

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 427499 โครงการศึกษาวิศวกรรมโทรคมนาคม

ประจำภาคการศึกษาที่ 2 ปีการศึกษา 2549

หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม

หลักสูตรปรับปรุงพ.ศ. 2546

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

โครงการ วงจรเชื่อมต่อคอมพิวเตอร์สำหรับช่วยสอนปฏิบัติการ
ผู้ดำเนินงาน นายวิรพงษ์ ห้าวหาญ B4608738
 นายอัฐพล เรืองบุตร B4611813
อาจารย์ที่ปรึกษา ผศ.ดร.รังสรรค์ วงศ์สรรค์
สาขาวิชา วิศวกรรมโทรคมนาคม
ภาคการศึกษาที่ 2/2549

บทคัดย่อ

โครงการนี้ได้ออกแบบจัดทำวงจรเชื่อมต่อคอมพิวเตอร์สำหรับช่วยสอนปฏิบัติการ เป็นการรับข้อมูลแล้วส่งผ่านข้อมูลไปแสดงผลยังจอคอมพิวเตอร์ สามารถแบ่งออกได้เป็น 2 ส่วนคือ วงจรเชื่อมต่อระหว่างแบบตัวอย่างของปฏิบัติการทางไฟฟ้ากับเครื่องคอมพิวเตอร์ และ โปรแกรมควบคุม โดยวงจรเชื่อมต่อนี้จะเป็นส่วนที่ทำหน้าที่ในการรับค่าสัญญาณพื้นฐานที่ได้จากการปฏิบัติการซึ่งเป็นค่าสัญญาณในรูปแบบอนาลอก และทำการแปลงค่าสัญญาณให้อยู่ในรูปแบบดิจิทัล เพื่อที่จะสามารถใช้เชื่อมต่อกับเครื่องคอมพิวเตอร์ได้ โดยในโครงการนี้จะได้ออกแบบตัวอย่างของปฏิบัติการทางไฟฟ้าเพื่อนำมาทดสอบการใช้งานของวงจรเชื่อมต่อที่ได้ออกแบบสร้างนี้ ในส่วนของโปรแกรมควบคุม จะมีการเขียนโปรแกรมควบคุมทางคอมพิวเตอร์ เพื่อแสดงผลที่ได้จากการวัด

กิตติกรรมประกาศ

ในการจัดทำโครงงานนี้จะมีอาจสำเร็จลุล่วงไปได้ด้วยดี ถ้าหากมิได้รับความกรุณาจากอาจารย์ที่ปรึกษา ผศ.ดร.รังสรรค์ วงศ์สรรค์ ผู้ที่ให้แนวคิดแรกเริ่มของวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ ที่ให้ความช่วยเหลือในเรื่องของแนวคิด การติดตามเอาใจใส่ และชี้แนะข้อบกพร่องที่ข้าพเจ้าได้มองข้ามไปในบางส่วน

นอกจากนี้ยังต้องขอขอบพระคุณผู้ที่เกี่ยวข้อง ดังนี้

คณาจารย์ประจำสาขาวิชาวิศวกรรมโทรคมนาคมทุกท่าน ที่ได้สั่งสอนให้ความรู้

คุณประพล จาระตะคุ ผู้ที่ให้ความช่วยเหลือในเรื่องของการเบิกจ่ายงบประมาณ ตลอดจนอุปกรณ์ในการทดลองต่างๆ

คุณณิรัตน์ ทุมพงษ์ เลขานุการประจำสาขาวิชาวิศวกรรมโทรคมนาคม ที่ให้ความช่วยเหลือในเรื่องของงานเอกสาร

พี่ๆ สาขาวิชาวิศวกรรมโทรคมนาคม ที่ให้ความช่วยเหลือในเรื่องของการเขียนโปรแกรม และการทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

สุดท้ายนี้ คุณงานความดีที่เกิดจากโครงงานฉบับนี้ ขอมอบแก่บิดามารดา ผู้ที่คอยห่วงใยให้กำลังใจ ให้โอกาส และให้การสนับสนุนทางการศึกษามาโดยตลอด

นายวีรพงษ์ ห้าวหาญ

นายอัฐพล เรืองบุตร

สารบัญ

	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญรูป	ช
สารบัญตาราง	ญ
 บทที่ 1 บทนำ	 1
1.1 ความเป็นมา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของการทำงาน	1
1.4 ขั้นตอนการดำเนินงาน	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
 บทที่ 2 ทฤษฎีและหลักการ	 3
2.1 ไมโครคอนโทรลเลอร์	3
2.1.1 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51	3
2.1.2 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ตระกูล MCS-51	4
2.1.3 การจัดขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51	4
2.2 USB (Universal Serial Bus)	6
2.2.1 ระบบการจัดส่งข้อมูล USB	6
2.3 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	7
2.3.1 ความหมายของสัญญาณอนาลอกและสัญญาณดิจิทัล	8
2.3.2 Analog to Digital Converter (A/D)	8
2.3.3 การมอดูเลตทางแอมพลิจูดของพัลส์	9
2.3.4 การควอนไทซ์	9
2.3.5 การแปลงระดับสัญญาณเป็นเชิงเลข	10

สารบัญ

	หน้า
2.4 วงจรแบ่งแรงดัน	11
2.5 สรุป	12
บทที่ 3 การออกแบบและการสร้างวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ	13
3.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (A/D)	13
3.1.1 การออกแบบลายวงจร A/D ลงบนแผ่นปริ้นท์	13
3.1.2 การเลือกใช้ชิพในการออกแบบ	14
3.1.3 การลงอุปกรณ์อิเล็กทรอนิกส์ตามที่ได้ออกแบบ	15
3.2 วงจรไมโครคอนโทรลเลอร์	15
3.3 วงจรเชื่อมต่อ USB	16
3.3.1 วงจรเชื่อมต่อ USB	16
3.3.2 การเลือกใช้ไอซีในการออกแบบ	17
3.3.3 การลงอุปกรณ์อิเล็กทรอนิกส์ตามที่ได้ออกแบบ	19
3.4 สรุปผลการออกแบบ	19
บทที่ 4 การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์สำหรับช่วยสอนปฏิบัติการ	20
4.1 การทดสอบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	20
4.1.1 การติดตั้งชุดวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	20
4.1.2 การเขียนโปรแกรมทดสอบ	20
4.1.3 การโหลดโปรแกรมทดสอบ	21
4.1.4 ทดสอบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	23
4.2 การทดสอบวงจรเชื่อมต่อ USB	26
4.2.1 การติดตั้งชุดวงจรเชื่อมต่อ USB เพื่อการทดสอบ	26
4.2.2 การติดตั้งไดรเวอร์วงจรเชื่อมต่อ USB	27
4.2.3 การทดสอบวงจรเชื่อมต่อ USB	31

สารบัญ

	หน้า
4.3 การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ	31
4.3.1 การติดตั้งชุดวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ	31
4.3.2 การเขียนโปรแกรมควบคุมและแสดงผล	32
4.3.3 การโหลดโปรแกรมเพื่อทดสอบ	32
4.3.4 การทดสอบวงจรเชื่อมต่อช่วยสอนปฏิบัติการและโปรแกรม	33
4.4 สรุปผลการทดสอบวงจร	37
4.4.1 การทดสอบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	37
4.4.2 การทดสอบวงจรเชื่อมต่อ USB	37
4.4.3 การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ	37
บทที่ 5 สรุปและข้อเสนอแนะ	38
5.1 ส่วนประกอบของวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ	38
5.2 ปัญหาที่พบในขั้นตอนการทำงาน	38
5.3 ข้อจำกัดของโครงงาน	38
5.4 ผลที่ได้จากโครงงาน	39
5.5 แนวทางการพัฒนาต่อ	39
บรรณานุกรม	40
ภาคผนวก (ก)	41
โปรแกรมควบคุมและสั่งการให้แสดงผลผ่านทางหน้าจอคอมพิวเตอร์ ของ “วงจรแบ่งแรงดัน”	42
โปรแกรมควบคุมการทำงานของวงจรแปลงสัญญาณอนาลอก เป็นสัญญาณดิจิทัล	45
แผนผังการทำงานของโปรแกรมควบคุม	48

สารบัญ

	หน้า
ภาคผนวก (ข)	50
อุปกรณ์ที่ใช้ และการลงอุปกรณ์บนแผ่นปริน	51
วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	51
วงจรเชื่อมต่อ USB	52
ประวัติผู้จัดทำ	54

สารบัญรูป

	หน้า
2.1 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-51	3
2.2 การจัดขาของ MCS-51	4
2.3 การเชื่อมต่อพอร์ต USB	7
2.4 เปรียบเทียบสัญญาณอนาลอกกับสัญญาณดิจิทัล	8
2.5 Analog - to – Digital	8
2.6 การทำ PAM สัญญาณอนาลอก	9
2.7 Quantized PAM Signal	9
2.8 Quantizing Using Sign and Magnitude	10
2.9 สัญญาณดิจิทัลที่ได้จากการทำ PCM สัญญาณอนาลอกเรียบร้อยแล้ว	10
2.10 PCM Process	10
2.11 (ก) วงจรแบ่งแรงดัน	11
2.11 (ข) วงจรแบ่งแรงดันซึ่งกำหนดค่ากระแส i	11
2.12 วงจรแบ่งแรงดันที่ต่อกับโหลด R_L	12
3.1 ภาพโครงสร้างโดยรวมของระบบ	13
3.2 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	13
3.3 ลายวงจรที่ได้จากโปรแกรม Protel99SE บนแผ่นปริ้นท์	14
3.4 (ก) Functional Diagram ของ ADS7841	14
3.4 (ข) Top View ของ ADS7841	14
3.5 ชุดวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล	15
3.6 ชุดวงจรไมโครคอนโทรลเลอร์ AT89C51ED2	15
3.7 วงจรเชื่อมต่อ USB	16
3.8 ลายวงจรที่ได้จากโปรแกรม Protel99SE บนแผ่นปริ้นท์	16
3.9 (ก) Data Flow	17
3.9 (ข) USB-to-Serial (Single Channel) Controller Block Diagram	17
3.10 Top View ของ TUSB3410	18
3.11 (ก) Functional Block Diagram	18
3.11 (ข) Top View ของ TPS79133	18

สารบัญรูป

	หน้า
3.12 (ก) ชุดวงจรเชื่อมต่อ USB ด้านบน	19
3.12 (ข) ชุดวงจรเชื่อมต่อ USB ด้านล่าง	19
4.1 วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลเพื่อการทดสอบ	20
4.2 แสดงโปรแกรม Keil70	20
4.3 แสดงการคอมไพล์สมบูรณ์	21
4.4 หน้าต่างโปรแกรม FLIP	21
4.5 หน้าต่างเลือกชนิดไมโครคอนโทรลเลอร์	22
4.6 ตั้งค่าการเชื่อมต่อ	22
4.7 โหลดไฟล์ .HEX	23
4.8 (ก) หน้าต่าง Hyper Terminal	24
4.8 (ข) การตั้งค่าอัตราการส่งข้อมูล	24
4.9 (ก) ป้อนแรงดัน 0 โวลต์	24
4.9 (ข) ป้อนแรงดัน 2.5 โวลต์	25
4.9 (ค) ป้อนแรงดัน 5 โวลต์	25
4.10 การต่อวงจรเชื่อมต่อ USB เพื่อทดสอบ	26
4.11 (ก) Driver Installation Progress	27
4.11 (ข) License agreement	27
4.11 (ค) Pre-installation of drivers complete	28
4.12 (ก) Found New Hardware (“TUSB3410 Device”)	28
4.12 (ข) Device installation complete	29
4.13 (ก) Found New Hardware (USB – Serial Port)	29
4.13 (ข) USB-Serial port installation complete	30
4.14 Windows XP Device Manager	30
4.15 ทดสอบส่งข้อมูลผ่านวงจรเชื่อมต่อ USB	31
4.16 วงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ	31
4.17 วงจรทดสอบ	32
4.18 ผลการทดสอบอินพุตอนาล็อกช่องที่ 1	33

สารบัญรูป

	หน้า
4.19 ผลการทดสอบอินพุตนอกช่องที่ 2	34
4.20 ผลการทดสอบอินพุตนอกช่องที่ 3	35
4.21 ผลการทดสอบอินพุตนอกช่องที่ 4	36

สารบัญตาราง

	หน้า
4.1 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตอนาล็อกช่องที่ 1	34
4.2 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตอนาล็อกช่องที่ 2	35
4.3 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตอนาล็อกช่องที่ 3	36
4.4 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตอนาล็อกช่องที่ 4	37

บทที่ 1

บทนำ

1.1 ความเป็นมา

ในปัจจุบันการใช้งานเครื่องคอมพิวเตอร์มีบทบาทสำคัญในการใช้งานในห้องปฏิบัติการมากขึ้นกว่าในสมัยก่อน และเครื่องมือที่ใช้ในการทดลองหรือวัดค่าต่าง ๆ มีราคาที่สูงขึ้นสูงในบางประเภท และความนิยมในการใช้งานเครื่องมือที่มีการติดต่อกับคอมพิวเตอร์ผ่านทางพอร์ต USB มีความแพร่หลาย ดังนั้นโครงการที่จัดทำขึ้นนี้ก็เพื่อเป็นการสร้างอุปกรณ์ ที่ช่วยในการเรียนรู้ในห้องปฏิบัติการให้ทำการแสดงผลที่ได้ออกมาผ่านทางหน้าจอคอมพิวเตอร์โดยติดต่อกับอุปกรณ์ผ่านทางพอร์ต USB

1.2 วัตถุประสงค์

- 1.2.1 เพื่อศึกษาและออกแบบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล
- 1.2.2 เพื่อพัฒนาโปรแกรมคอมพิวเตอร์ใช้ทำการประมวลผลและแสดงผลได้ตามที่กำหนด
- 1.2.3 เพื่อศึกษาออกแบบวงจรเชื่อมต่อคอมพิวเตอร์จาก RS-232 ไปเป็น USB

1.3 ขอบเขตของการทำงาน

- 1.3.1 ศึกษาค้นหาว่าข้อมูลเกี่ยวกับพอร์ต USB, การแปลงสัญญาณอนาลอกเป็น สัญญาณดิจิทัล
- 1.3.2 ออกแบบวงจรเชื่อมต่อเพื่อใช้งานระหว่าง แบบตัวอย่างของปฏิบัติการทางไฟฟ้ากับเครื่องคอมพิวเตอร์
- 1.3.3 เขียนโปรแกรมควบคุมวงจรเชื่อมต่อสัญญาณเพื่อให้อ่านค่าที่ได้
- 1.3.4 เขียนโปรแกรมประมวลผลค่าการวัดเพื่อใช้แสดงผลผ่านหน้าจอคอมพิวเตอร์
- 1.3.5 สร้างอุปกรณ์ต้นแบบทั้งหมดเพื่อทำการทดสอบเพื่อให้ได้ตามวัตถุประสงค์

1.4 ขั้นตอนการดำเนินงาน

- 1.4.1 ศึกษาและทำการค้นคว้าเกี่ยวกับการทำงานของพอร์ต USB และการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล
- 1.4.2 ออกแบบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

- 1.4.3 ออกแบบวงจรเชื่อมต่อ คอมพิวเตอร์กับไมโครคอนโทรลเลอร์ ผ่านทางพอร์ต USB
- 1.4.4 จัดเตรียมวัสดุและอุปกรณ์ต่างๆ
- 1.4.5 จัดทำและประกอบชิ้นส่วนของอุปกรณ์ต่าง ๆ ตามที่ได้ออกแบบไว้
- 1.4.6 เขียนโปรแกรมเพื่อให้ไมโครคอนโทรลเลอร์สั่งการให้วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลทำงานและเกิดการเชื่อมต่อข้อมูลกับคอมพิวเตอร์ ผ่านทางพอร์ต USB
- 1.4.7 ทดสอบและแก้ไขการทำงานของระบบที่ได้ออกแบบ
- 1.4.8 สรุปผลการทดลองและประเมินผลและจัดทำรูปเล่มรายงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1.5.1 ได้เรียนรู้การทำงานเป็นทีม การรู้จักรับฟังความคิดเห็นของเพื่อนร่วมงาน
- 1.5.2 ได้เรียนรู้วิธีหาความรู้ด้วยตัวเองเพื่อนำมาปฏิบัติและประยุกต์ใช้งานจริง
- 1.5.3 ได้เรียนรู้วิธีและกระบวนการแก้ปัญหาต่างๆที่เกิดขึ้นอย่างมีประสิทธิภาพ
- 1.5.4 สามารถพัฒนาทักษะการใช้โปรแกรมที่เกี่ยวข้องกับโครงงานให้มีความชำนาญขึ้น
- 1.5.5 ได้รับความรู้มากขึ้นในทางปฏิบัติหลังจากทำการศึกษาทางทฤษฎีมาแล้ว
- 1.5.6 สามารถค้นคว้าหาความรู้เพิ่มเติมนอกเหนือจากหลักสูตรที่เรียนได้
- 1.5.7 ได้บรรลุเป้าหมายตามวัตถุประสงค์ที่วางไว้ทั้งหมด

บทที่ 2

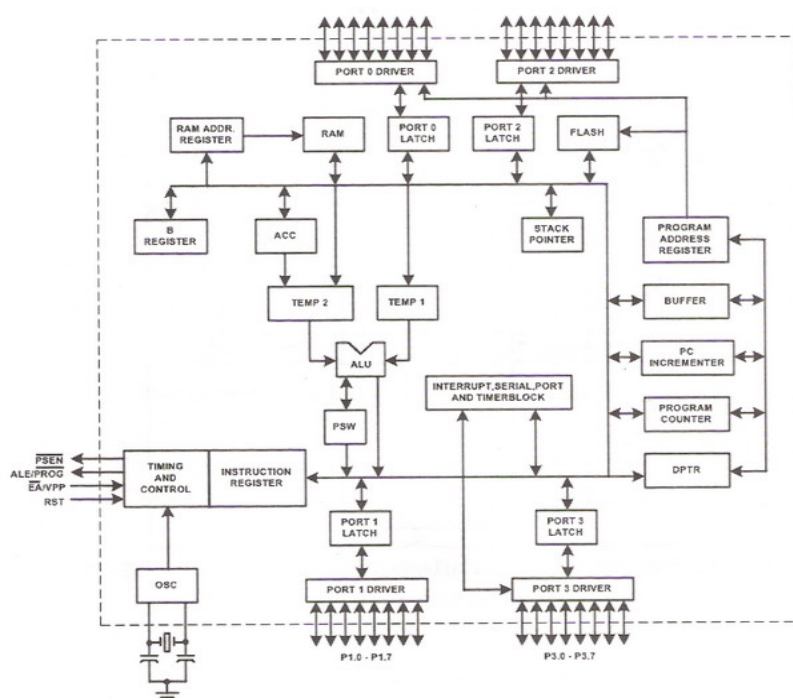
ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์ MCS-51 [1]

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้จัดให้มีส่วนประกอบภายในเพื่ออำนวยความสะดวกแก่ผู้ใช้ เช่น ไทเมอร์ (Timer) /เคาเตอร์ (Counter), พอร์ตอนุกรม (Serial port) และสำหรับไมโครคอนโทรลเลอร์เบอร์ใหญ่ ๆ ยังอาจมีส่วนอื่นเพิ่มเติมเข้ามาอีก เช่น เบอร์ 80C515, 80C535 จะมีวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

2.1.1 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ใช้เทคโนโลยีการผลิตแบบ NMOS และ CMOS ซึ่งภายในได้รวมวงจรต่างไว้อย่างครบถ้วนพร้อมที่จะทำงานได้เมื่อจ่ายไฟเลี้ยงและสัญญาณนาฬิกา (Clock) ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้ถูกผลิตออกมามากมายหลายเบอร์โดยบริษัทต่าง ๆ ไม่ว่าจะเป็นเบอร์อะไรก็ตาม ถ้าเป็นไมโครคอนโทรลเลอร์ตระกูล MCS-51 แล้วจะมีโครงสร้างต่าง ๆ ที่คล้ายกัน จะต่างกันออกไปในส่วนของความสามารถพิเศษของแต่ละเบอร์ ยกตัวอย่างเช่นเบอร์ AT89C51 มีไทเมอร์ 2 ตัว ในขณะที่เบอร์ AT89C52 มีไทเมอร์ 3 ตัว เป็นต้น โดยที่โครงสร้างต่างๆ ภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้แสดงไว้ดังรูปที่ 2.1



รูปที่ 2.1 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-51

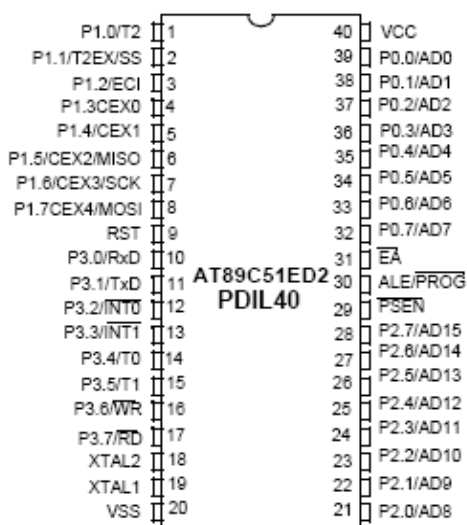
2.1.2 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ตระกูล MCS-51

เนื่องจากคุณสมบัติของไมโครคอนโทรลเลอร์แต่ละเบอร์นั้นมีความสามารถที่แตกต่างกันออกไปในรายละเอียดปลีกย่อย ดังนั้นจะขออ้างถึงเบอร์ AT89C51 ของบริษัท Atmel ซึ่งเป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ซึ่งมีคุณสมบัติดังนี้

- มีหน่วยความจำโปรแกรมแบบแฟลช (Flash Memory) ขนาด 8 กิโลไบต์
- มีหน่วยความจำแบบ EEPROM ขนาด 2 กิโลไบต์ (kbyte)
- แหล่งจ่ายไฟกระแสตรงขนาด 5 โวลต์ (ทำงานในช่วง 4-6 โวลต์)
- ทำงานได้ด้วยสัญญาณนาฬิกาตั้งแต่ 0-24 เมกกะเฮิร์ต (MHz)
- มีหน่วยความจำข้อมูล (RAM) ขนาด 256 ไบต์ (byte)
- มีพอร์ต 32 พอร์ตอิสระสามารถเข้าถึงในระดับบิตได้
- มีไทมเมอร์/เคานเตอร์ขนาด 16 บิต ทั้งหมด 3 ตัว
- รองรับการอินเทอร์รัพต์ได้ 8 แหล่ง
- สามารถสื่อสารข้อมูลแบบอนุกรมได้ด้วย UART Channel

2.1.3 การจัดขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ในตระกูล MCS-51 จะมีโครงสร้างของการจัดเรียงขาที่คล้าย ๆ กันได้แสดงไว้ดังรูปที่ 2.2 และมีส่วนประกอบของขาดังต่อไปนี้



รูปที่ 2.2 การจัดขาของ MCS-51

VCC ต่อไฟเลี้ยง

GND ต่อกราวด์

Port0 (P0.0-P0.7) เป็นพอร์ตแบบสองทิศทางขนาด 8 บิต สามารถทำงานได้ทั้งสองหน้าที่คือ เป็นพอร์ตอินพุตและเอาต์พุตทั่วไป และใช้เป็นพอร์ตสำหรับติดต่อกับหน่วยความจำภายนอกคือรับและส่งข้อมูลพร้อมทั้งกำหนดแอดเดรส (Address) ไปด้ต่ำ

Port1 (P1.0-P1.7) เป็นพอร์ตแบบสองทิศทางขนาด 8 บิต มีการต่อความต้านทานพูลอัพ (pull-up resistor) ไว้ภายใน ทำหน้าที่เป็นพอร์ตอินพุตและเอาต์พุตทั่วไป นอกจากนี้ยังใช้งานเป็นขาอินพุตเอาต์พุตของไทมเมอร์ 2

Port2 (P2.0-P2.7) เป็นพอร์ตแบบสองทิศทางขนาด 8 บิต มีการต่อความต้านทานพูลอัพ (pull-up resistor) ไว้ภายใน สามารถทำงานได้สองหน้าที่คือเป็นพอร์ตอินพุตและเอาต์พุตทั่วไป และใช้เป็นพอร์ตสำหรับติดต่อกับหน่วยความจำภายนอกคือกำหนดแอดเดรสไบต์สูง

Port3 เป็นพอร์ตแบบสองทิศทางขนาด 8 บิต มีการต่อความต้านทานพูลอัพ (pull-up resistor) ไว้ภายใน ทำหน้าที่เป็นพอร์ตอินพุตและเอาต์พุตทั่วไป นอกจากนี้ยังใช้เป็นขาสัญญาณควบคุมการติดต่อกับหน่วยความจำการอินเทอร์รัปต์ และอื่น ๆ

RST เป็นขาอินพุตที่ไว้รับสัญญาณสำหรับรีเซ็ตชิพ โดยชิพ (CPU) จะถูกรีเซ็ต(Reset) เมื่อขานี้เป็นลอจิก “1” นาน 2 แมกซ์ไซเคิล หรือ 24 ไซเคิลของสัญญาณนาฬิกา

ALE/PROG ทำหน้าที่เป็นขาเอาต์พุตเมื่อชิพต้องการติดต่อกับหน่วยความจำภายนอก คือ จะทำการส่งสัญญาณพัลส์ (Pulse) ออกมาที่ขาเพื่อทำการแลกแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก และขาจะเป็นอินพุตเมื่ออยู่ในระหว่างโปรแกรมแฟลช

PSEN เป็นขาเอาต์พุต ใช้ในการติดต่อกับหน่วยความจำภายนอก คือเมื่อชิพทำการประมวลผลกับหน่วยความจำโปรแกรมภายนอกขานี้จะแอคทีฟสองครั้งในแต่ละแมกซ์ไซเคิล

EA/VPP เป็นขาอินพุตและต้องการลอจิก “0” เพื่อยอมให้ชิพสามารถเข้าถึงหน่วยความจำโปรแกรมภายนอกได้ ซึ่งอยู่ที่ตำแหน่ง 0000H ถึง FFFFH นอกจากนี้แล้วขานี้ยังใช้รับไฟ 12 โวลต์เพื่อใช้ในระหว่างที่ทำการโปรแกรมแฟลช

XTAL1 เป็นขาอินพุตของวงจรออสซิลเลเตอร์แอมพลิไฟเออร์ และยังเป็นขาอินพุตของวงจรกำเนิดสัญญาณนาฬิกาภายใน

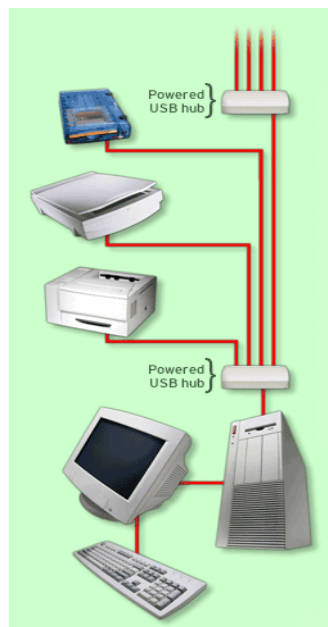
XTAL2 เป็นขาเอาต์พุตของวงจรออสซิลเลเตอร์แอมพลิไฟเออร์

2.2 USB (Universal Serial Bus USB) [2]

การสื่อสารระหว่างคอมพิวเตอร์และอุปกรณ์ต่อพ่วงต่างๆ จำเป็นที่จะต้อง มี พอร์ต I/O เพื่อใช้ในการสื่อสาร (I/O = input , output) ในสมัยก่อน พอร์ตที่รู้จักกันดี ก็คือ พอร์ตอนุกรม (Serial Port) หรือเรียกกันว่า พอร์ตคอม (Com Port) และ พอร์ตขนาน (Parallel Port) แต่ข้อจำกัดของพอร์ต เหล่านี้ นั้นมีมากพอสมควรไม่ว่าจะเป็นเรื่องของความเร็ว จำนวนอุปกรณ์จะนำมาต่อได้ และคุณสมบัติทางด้าน Plug & Play USB ก็เป็น พอร์ต I/O ที่ทำงานในลักษณะซีเรียลบัส (Serial Bus) เช่นกัน สามารถต่อกับอุปกรณ์ถึง 127 อุปกรณ์ ต่อ 1 พอร์ต และ สนับสนุนการทำงานแบบ Plug & Play อย่างเต็มรูปแบบ โดยจะทำการค้นหา และตั้งค่าของอุปกรณ์ที่นำมาต่อให้อย่างอัตโนมัติ ไม่จำเป็นต้องกำหนดค่า (assign) IRQ เพียงแต่ใส่แผ่นไดรเวอร์ (Driver) ของอุปกรณ์ที่นำมาติดตั้ง และส่วนมากไม่จำเป็นต้องปิดแล้วเปิดเครื่องใหม่ (Restart) หลังจากติดตั้งไดรเวอร์ด้วย ถ้าหากอุปกรณ์นั้นใช้กำลังไฟฟ้าไม่มาก พอร์ต USB ก็จ่ายไฟเลี้ยงให้กับอุปกรณ์ต่างๆ ได้ด้วย

2.2.1 ระบบการจัดส่งข้อมูล USB

สำหรับ USB เราเรียกระบบบัสเช่นนี้ว่า การติดต่อแบบ "Host/Slave" หมายถึงตัวเครื่องคอมพิวเตอร์จะเป็นผู้จัดการ การจัดส่งข้อมูลทั้งหมด และอุปกรณ์ที่นำมาต่อพ่วงเพียงแต่พร้อมที่จะรับส่งข้อมูลเท่านั้น โดยตัว USB Host Controller หรือตัวควบคุมของ USB นั้น จะรวมอยู่ในตัวชิพเซต (Chipset) บนเมนบอร์ด แต่ถ้าหากเป็นชิพเซตรุ่นเก่าๆ ก็ยังไม่มีตัวควบคุม USB ในการจัดส่งข้อมูลนั้น ภายในสาย USB จะมีสายภายในทั้งหมด 4 เส้น ในจำนวนนี้ 2 เส้นใช้เพื่อเป็นสายในการส่งข้อมูล อีก 2 เส้นใช้สำหรับจ่ายไฟเลี้ยง แรงดันบวก 5 โวลต์ และ กราวด์ ข้อมูลที่ถูกส่งผ่าน USB นั้นจะต่างไปจากพอร์ตขนาน และพอร์ตอนุกรมซึ่งทั้ง 2 ชนิดนี้นั้น จะส่งข้อมูลเป็นบิตเดี่ยวๆ แต่ USB จะส่งเป็นชุดข้อมูล อย่างเช่น ถ้าหากต้องการ เก็บข้อมูลไปยัง USB Zip drive PC จะทำการแบ่งข้อมูลออกมา 64 ไบต์ แล้วใส่ข้อมูลของแอดเดรสเข้าไปแล้วส่งไปยังพอร์ต USB จากนั้นก็จะทำเช่นนี้จนกระทั่งส่งข้อมูลครบสิ่งที่ทำให้ USB ส่งข้อมูลได้เร็วถึง 12 เมกกะบิตต่อวินาที (Mbps) นั้นอยู่ที่แม่นยำของการส่ง เพราะ สายที่นำมาใช้สำหรับ USB นั้น มีระดับสัญญาณรบกวน และ ความเพี้ยนของรูปสัญญาณนั้นน้อยกว่า ทั้งพอร์ตขนาน และพอร์ตอนุกรม และการส่งสัญญาณแบบ isochronous data delivery หมายถึง อุปกรณ์แต่ละชิ้นบนระบบบัสนั้น จะถูกจำกัดขนาดของแบนด์วิดท์ (bandwidth) ให้แน่นอน อย่างไรก็ตามแบนด์วิดท์โดยรวมแล้วก็ไม่เกิน 12 เมกกะบิตต่อวินาที นี่เป็นเหตุผลที่ทำให้ USB สามารถต่ออุปกรณ์ได้มากที่สุด 127 อุปกรณ์



รูปที่ 2.3 การเชื่อมต่อพอร์ต USB

จากรูปที่ 2.3 เป็นการต่ออุปกรณ์ต่อพ่วงออกจากเครื่องคอมพิวเตอร์ไปยังอุปกรณ์ต่าง ๆ เช่น จากพอร์ต USB แรกไปยัง Powered USB Hub แล้วต่อไปยังอุปกรณ์อื่น ๆ รวมถึงต่อไปยัง Powered USB Hub เพื่อต่ออุปกรณ์เพิ่มเข้าอีกได้เรื่อย ๆ ในลักษณะของการต่อแบบอนุกรม ส่วนพอร์ต USB ชุดที่ 2 หลังเครื่องนั้นสามารถต่อไปยังจอมอนิเตอร์ (Monitor) ได้โดยอิสระต่อกัน จะสังเกตได้ว่าถ้าหากการต่อมี USB Hub เข้ามาต่อพ่วงด้วยที่ตัว Hub จะต้องมีตัวจ่ายไฟด้วยเพราะถ้าพึ่งไฟเลี้ยงจากเครื่องคอมพิวเตอร์เองคงไม่พอส่งไปเลี้ยงอุปกรณ์ต่าง ๆ

ดังนั้นเราอาจมองได้ว่าสิ่งที่ต่อจากพอร์ต USB นั้นมี 2 ชนิด ก็คือ ต่อเข้าไปได้กับ Hubs หรือต่อกับ อุปกรณ์ Hubs นั้นมีหน้าที่ ทำให้ สามารถต่ออุปกรณ์ได้มากขึ้นบนระบบบัส บางครั้ง Hubs จะเข้าไป รวมกับอุปกรณ์ อย่างเช่น มอนิเตอร์ ที่มี USB Hubs ในตัว โดยจะ สามารถปรับค่า ต่างๆของมอนิเตอร์ผ่านระบบปฏิบัติการ และ นอกจากนั้นก็มีหน้าที่เป็น Hubs ด้วย

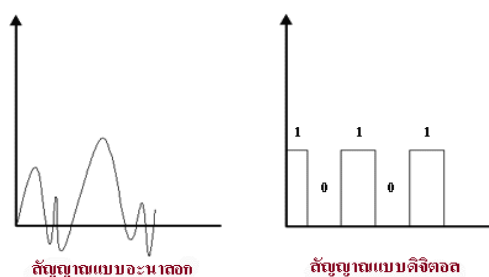
2.3 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล [3]

การสื่อสารข้อมูลด้วยคอมพิวเตอร์สามารถสื่อสารข้อมูลได้ทุกประเภท ประกอบด้วย เสียง (Voice) อักษรข้อความ (Text), ภาพ (Image) และข้อมูลคอมพิวเตอร์ (Data) ซึ่งแต่ละข้อมูล มีลักษณะเฉพาะของสัญญาณที่แตกต่างกัน แบ่งการกระทำของข้อมูลดังนี้

2.3.1 ความหมายของสัญญาณอนาลอกและสัญญาณดิจิทัล

สัญญาณอนาลอก คือ สัญญาณข้อมูลแบบต่อเนื่อง (Continuous Data) มีขนาดของสัญญาณไม่คงที่ การเปลี่ยนแปลงขนาดของสัญญาณแบบค่อยเป็นค่อยไปแปรผันตามเวลา การส่งสัญญาณแบบอนาลอก จะถูกรบกวนให้มีการแปลความหมายผิดพลาดได้ง่ายสัญญาณแบบอนาลอกนี้จะเป็นสัญญาณที่สื่อกลางในการสื่อสารส่วนมากใช้อยู่ เช่น สัญญาณเสียงในสายโทรศัพท์

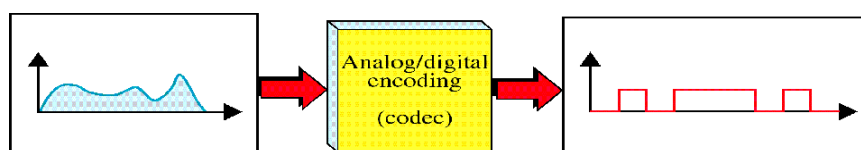
สัญญาณดิจิทัล คือ สัญญาณข้อมูลแบบไม่ต่อเนื่อง (Discrete Data) มีขนาดของสัญญาณคงที่ การเปลี่ยนแปลงขนาดของสัญญาณเป็นแบบทันทีทันใด ไม่แปรผันตามเวลามีประสิทธิภาพและความน่าเชื่อถือสูงกว่าแบบอนาลอก เนื่องจากการใช้งานเพียง 2 ค่าเพื่อนำมาตีความหมายเป็น on / off หรือ 1 / 0 เท่านั้น ซึ่งสัญญาณดิจิทัลนี้จะเป็นสัญญาณที่ คอมพิวเตอร์ใช้ในการทำงานและติดต่อสื่อสาร โดยได้แสดงการเปรียบเทียบความแตกต่างของสัญญาณอนาลอกกับสัญญาณดิจิทัล ดังรูปที่ 2.4



รูปที่ 2.4 เปรียบเทียบสัญญาณอนาลอกกับสัญญาณดิจิทัล

2.3.2 Analog to Digital Converter (A/D)

ทำหน้าที่แปลงสัญญาณข้อมูลที่มนุษย์รับรู้สัมผัสได้ เป็นข้อมูลทางไฟฟ้า เพื่อป้อนเข้าสู่การประมวลผล จึงเป็นขบวนการหนึ่งของการรับข้อมูล (Input Unit) เป็นกระบวนการอิเล็กทรอนิกส์ ที่สัญญาณที่ต่อเนื่อง (อนาลอก) ได้รับการแปลงให้เป็นสัญญาณที่ไม่ต่อเนื่อง (ดิจิทัล) ดังรูปที่ 2.5 โดยไม่มีการลบข้อมูลสำคัญ ผลลัพธ์ของ ADC มีลักษณะตรงข้าม คือ กำหนดระดับหรือสถานะ ตัวเลขของสถานะมักจะเป็นการยกกำลังของ 2 คือ 2, 4, 8, 16 เป็นต้น สัญญาณดิจิทัลพื้นฐานมี 2 สถานะและเรียกว่า ไบนารี (Binary) ตัวเลขทั้งหมดสามารถแสดงในรูปของไบนารี ในฐานะข้อความของ หนึ่งและศูนย์



รูปที่ 2.5 Analog - to - Digital

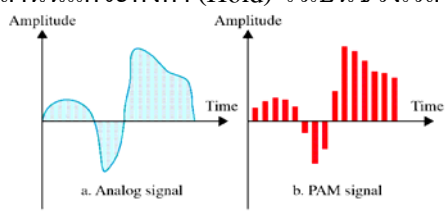
ส่วนใหญ่ในการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลจะใช้วิธีการมอดูเลตแบบรหัสพัลส์ (PCM: Pulse Code Modulation) ในการแปลงสัญญาณวิธีโดยการสุ่มช่วงเวลาใดเวลาหนึ่งทำให้เป็นไม่ต่อเนื่องกัน โดยให้ขนาด (Amplitude) ของสัญญาณเท่าเดิม (ถ้าทำการชักตัวอย่าง (sampling) ถี่มากก็จะเป็นต่อเนื่องเหมือนเดิม) ควอนไทซิง (Quantizing) ทำให้สัญญาณเป็นสัญญาณดิจิทัลโดยสมบรูณ์ก็จะต้องใส่แรงดัน (Voltage) เข้าไปในแต่ละระดับ (Levels) ของพัลส์ (Pulses)

ในขั้นตอนของการทำการมอดูเลตแบบรหัสพัลส์นั้นจะประกอบไปด้วย 4 กระบวนการคือ

- การมอดูเลตทางแอมพลิจูดของพัลส์ (PAM: Pulse Amplitude Modulation)
- การควอนไทซ์ (Quantization)
- การแปลงระดับสัญญาณเป็นเชิงเลข (Digitization)
- การทำไลน์โค้ดดิ้ง (Line Coding)

2.3.3 การมอดูเลตทางแอมพลิจูดของพัลส์

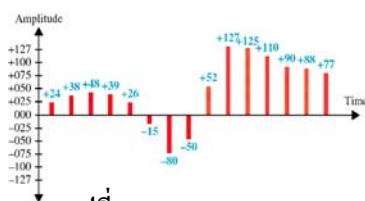
การมอดูเลตทางแอมพลิจูดของพัลส์ คือ การเอาข้อมูลที่เป็นอนาลอกมาทำการสุ่มสัญญาณเป็นช่วงที่มีระยะเท่าๆ กัน (Sampling and Hold) ค่าที่ได้จากการสุ่ม (Sample) จะมาจากฟังก์ชันเดลตา (Delta Function) ณ เวลานั้นแต่จะคงค่า (Hold) ไว้เป็นช่วงเวลาสั้นๆ แสดงได้ดังรูปที่ 2.6



รูปที่ 2.6 การทำ PAM สัญญาณอนาลอก

2.3.4 การควอนไทซ์

การควอนไทซ์ คือ การกำหนดค่าตัวเลขจำนวนเต็มให้กับค่าที่อ่านได้จากการทำการมอดูเลตทางแอมพลิจูดของพัลส์ เพื่อให้ทราบว่าสัญญาณที่ผ่านการมอดูเลตทางแอมพลิจูดของพัลส์แล้วนั้น อยู่ในระดับใด แสดงได้ดังรูปที่ 2.7



รูปที่ 2.7 Quantized PAM Signal

2.3.5 การแปลงระดับสัญญาณเป็นเชิงเลข

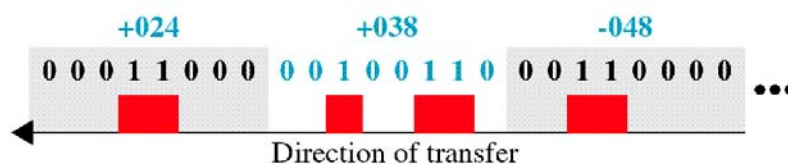
การแปลงระดับสัญญาณเป็นเชิงเลข คือ การแปลงระดับสัญญาณที่เป็นจำนวนเต็ม ให้เป็นอนุกรมของเลขฐานสอง โดยที่ระดับสัญญาณหาได้จากผลคูณถ่วงน้ำหนัก (ตำแหน่ง) ของเลขฐานสอง (ไม่นับบิตเครื่องหมาย)

$$L = \sum_{i=0}^{N-1} B_i * 2^i$$

+024	00011000	-015	10001111	+125	01111101
+038	00100110	-080	11010000	+110	01101110
+048	00110000	-050	10110010	+090	01011010
+039	00100111	+052	00110110	+088	01011000
+026	00011010	+127	01111111	+077	01001101

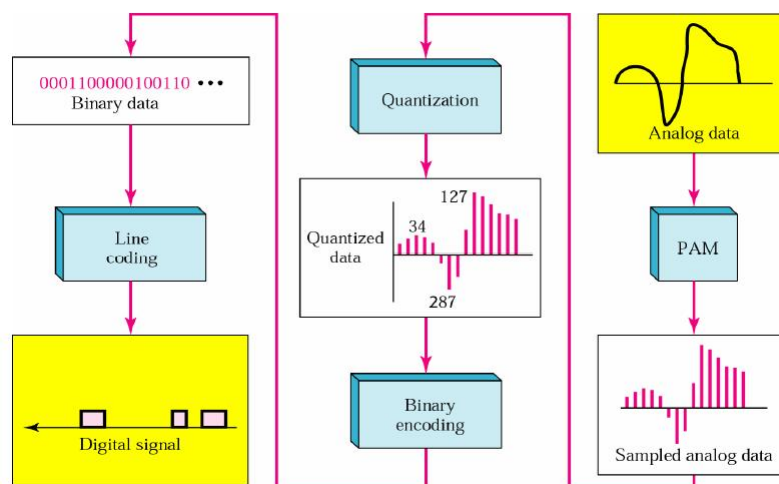
Sign bit
+ is 0 - is 1

รูปที่ 2.8 Quantizing Using Sign and Magnitude



รูปที่ 2.9 สัญญาณดิจิทัลที่ได้จากการทำ PCM สัญญาณอนาลอกเรียบร้อยแล้ว

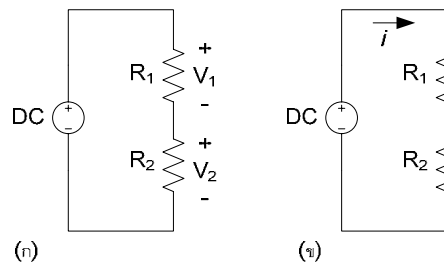
จากรูปที่ 2.8 และรูปที่ 2.9 เป็นการนำเอาระดับสัญญาณที่ผ่านการแปลงเป็นจำนวนเต็มแล้วมาทำการเข้ารหัสให้เป็นสัญญาณดิจิทัล และรูปที่ 2.10 เป็นการแสดงภาพรวมการทำการมอดูเลตทางแอมพลิจูดของพัลส์ของสัญญาณอนาลอก



รูปที่ 2.10 PCM Process

2.4 วงจรแบ่งแรงดัน [4]

ในวงจรแบ่งแรงดันเราสามารถวิเคราะห์ห้วงจรโดยใช้กฎของโอห์ม และกฎของเคอร์ชอฟฟ์เพื่อช่วยในการวิเคราะห์ เราจะได้กำหนดกระแส (i) ดังแสดงในรูปที่ 2.11 จากกฎกระแสของเคอร์ชอฟฟ์ กระแสที่ผ่าน R_1 และ R_2 มีค่าเท่ากัน จากการใช้กฎแรงดันของเคอร์ชอฟฟ์รอบทางเดินวงรอบปิด จะได้สมการ



รูปที่ 2.11 (ก) วงจรแบ่งแรงดัน

(ข) วงจรแบ่งแรงดันซึ่งกำหนดค่ากระแส i

$$v_s = iR_1 + iR_2 \quad (2.1)$$

หรือ

$$i = \frac{v_s}{R_1 + R_2} \quad (2.2)$$

เราสามารถใช้อัตราส่วนของโอห์มเพื่อคำนวณหาค่า v_1 และ v_2 :

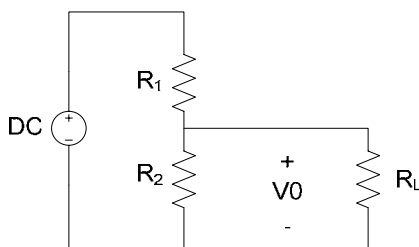
$$v_1 = iR_1 = v_s \frac{R_1}{R_1 + R_2} \quad (2.3)$$

$$v_2 = iR_2 = v_s \frac{R_2}{R_1 + R_2} \quad (2.4)$$

สมการที่ (2.3) และ (2.4) แสดงให้เห็นว่า v_1 และ v_2 คืออัตราส่วนของ v_s โดยที่อัตราส่วนแต่ละค่า คืออัตราส่วนความต้านทานที่ต้องการหาค่าแรงดันที่ตกคร่อมตัวมันอยู่ต่อผลบวกของค่าความต้านทานทั้งสอง เนื่องจากอัตราส่วนทั้งสองค่านี้มีค่าน้อยกว่า 1.0 เสมอ ค่าแรงดัน v_1 และ v_2 จึงมีค่าน้อยกว่าค่าแรงดัน v_s จากแหล่งกำเนิด

ถ้าเราต้องการให้ v_s มีค่าๆ หนึ่งและทราบค่าของ v_s ซึ่งจะมีค่า R_1 และ R_2 จำนวนอนันต์ที่ให้ค่า v_s ที่ต้องการ ตัวอย่างเช่น สมมติว่า v_s เท่ากับ 15 โวลต์ และ v_s ที่ต้องการคือ 5 โวลต์ ดังนั้น $v_2 / v_s = 1/3$ จากสมการที่ (2.4) จะเห็นว่า $R_2 = \frac{1}{2} R_1$ ปัจจัยอื่นๆ ที่ต้องคำนึงถึงในการเลือกค่า R_1 และ R_2 คือกำลังไฟฟ้าสูญเสียที่เกิดขึ้นจากการแบ่งแรงดันจากแหล่งกำเนิด และผลของการต่อวงจรแบ่งแรงดันกับอุปกรณ์อื่น ๆ ในวงจร

พิจารณาการต่อ R_L ขนานกับ R_2 ดังวงจรแสดงในรูปที่ 2.12 ตัวต้านทาน R_L ทำหน้าที่เสมือนโหลดของวงจรแบ่งแรงดัน โหลดของวงจรใดๆ สามารถประกอบด้วยอุปกรณ์ไฟฟ้าเพียงตัวเดียวหรือมากกว่าซึ่งดึงกำลังไฟฟ้าจากวงจร เมื่อตัวต้านทาน R_L ต่อในวงจรแบ่งแรงดัน



รูปที่ 2.12 วงจรแบ่งแรงดันที่ต่อกับโหลด R_L

เอาท์พุตสามารถหาได้จาก

$$v_0 = \frac{R_{eq}}{R_1 + R_{eq}} v_s \quad (2.5)$$

เมื่อ

$$R_{eq} = \frac{R_2 R_L}{R_2 + R_L} \quad (2.6)$$

แทนสมการที่ (2.6) ลงในสมการที่ (2.5) จะได้

$$v_0 = \frac{R_2}{R_1[1 + (R_2 / R_L)] + R_2} v_s \quad (2.7)$$

จะเห็นว่าสมการที่ (2.7) จะกลายเป็นสมการที่ (2.4) เมื่อ $R_L \rightarrow \infty$ สมการที่ (2.7) แสดงให้เห็นว่าถ้า $R_L \gg R_2$ อัตราส่วนแรงดัน v_0 / v_s ไม่เปลี่ยนแปลงไปมากนัก จากการเพิ่มโหลดในวงจรแบ่งแรงดัน

ลักษณะอีกอย่างของวงจรแบ่งแรงดันคือ ความไวของการแบ่งแรงดันต่อค่าความคลาดเคลื่อนที่ยอมรับได้ของตัวต้านทาน ค่าความต้านทานของตัวต้านทานที่มีขายในปัจจุบันจะมีความคลาดเคลื่อนตามจำนวนร้อยละที่ระบุ

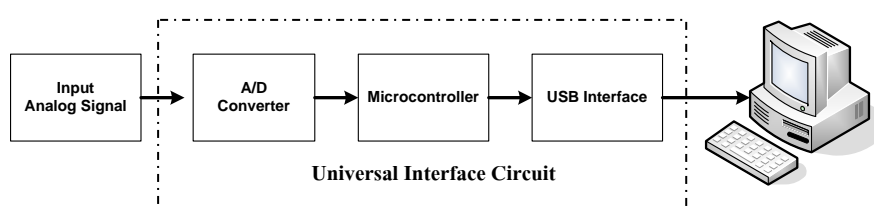
2.5 สรุป

วงจรเชื่อมต่อกับคอมพิวเตอร์ช่วยสอนปฏิบัติการมีองค์ประกอบที่สำคัญคือ วงจรแปลงสัญญาณแอนะล็อกไปเป็นสัญญาณดิจิทัล, วงจรเชื่อมต่อ USB และวงจรโครคอนโทรลเลอร์ โดยที่ทั้ง 3 ส่วนจะต้องทำงานร่วมกันอย่างเป็นระบบด้วยการเขียนโปรแกรมเพื่อทำการควบคุมการทำงาน

บทที่ 3

การออกแบบและการสร้างวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

ระบบโดยรวมของวงจรเชื่อมต่อมีอยู่ทั้งหมด 3 ส่วนด้วยกันโดยได้แสดงไว้ดังรูปที่ 3.1 คือ วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล, วงจรไมโครคอนโทรลเลอร์, วงจรเชื่อมต่อพอร์ต USB

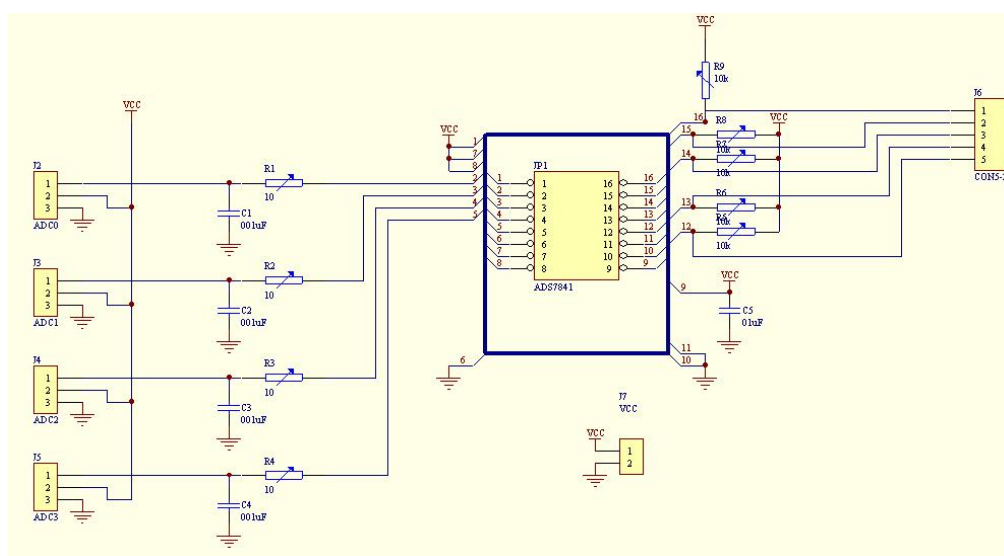


รูปที่ 3.1 ภาพโครงสร้างโดยรวมของระบบ

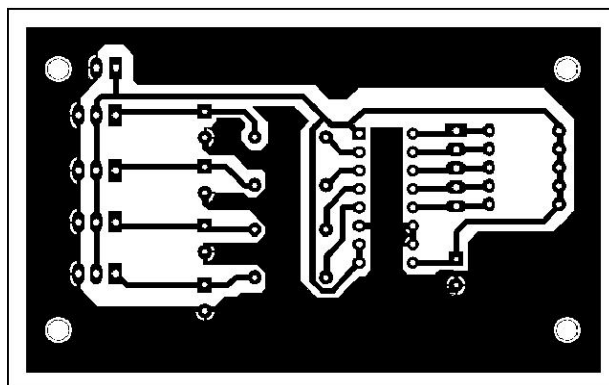
3.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (A/D)

3.1.1 การออกแบบลายวงจร A/D ลงบนแผ่นปริ้นท์ (PCB)

จากรูปที่ 3.2 ได้แสดงถึงวงจรที่ได้ทำการออกแบบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลไว้ โดยเลือกใช้ไอซีที่มีคุณสมบัติตามต้องการ จากนั้นทำการออกแบบลายวงจรด้วยโปรแกรม Protel 99SE [4] เพื่อให้ได้ลายวงจรเป็นแผ่นลายทองแดงที่จะสามารถนำมาใช้งานได้ตามรูปที่ 3.3



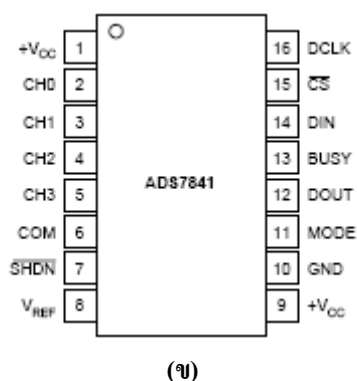
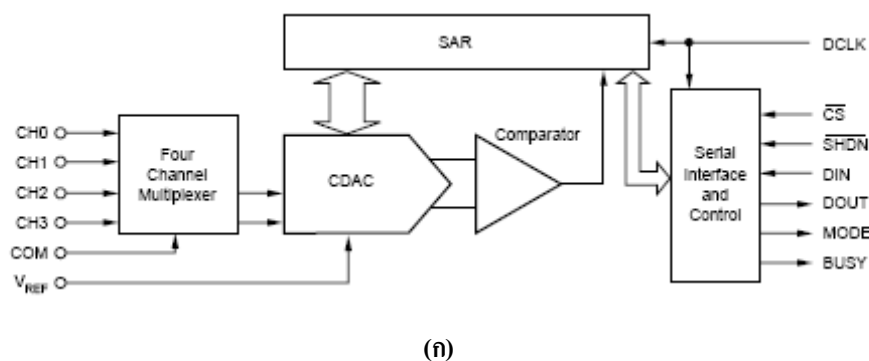
รูปที่ 3.2 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล



รูปที่ 3.3 ลายวงจรที่ได้จากโปรแกรม Protel99SE บนแผ่นปริ้นท์

3.1.2 การเลือกใช้ไอซีในการออกแบบ

ในวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลนั้นได้เลือกใช้ ไอซี เบอร์ **ADS7841** ในการแปลงสัญญาณ ซึ่งมีฟังก์ชันการทำงานและการจัดเรียงขาของไอซี ดังรูปที่ 3.4(ก) และรูปที่ 3.4 (ข) ตามลำดับ โดยที่ไอซีนี้สามารถรับสัญญาณอินพุตอนาลอกได้ทั้งหมด 4 ช่อง มีอัตราในการสุ่มเลือกตัวอย่าง 200 กิโลเฮิร์ต มีความแม่นยำในการรับสัญญาณ 12 บิต

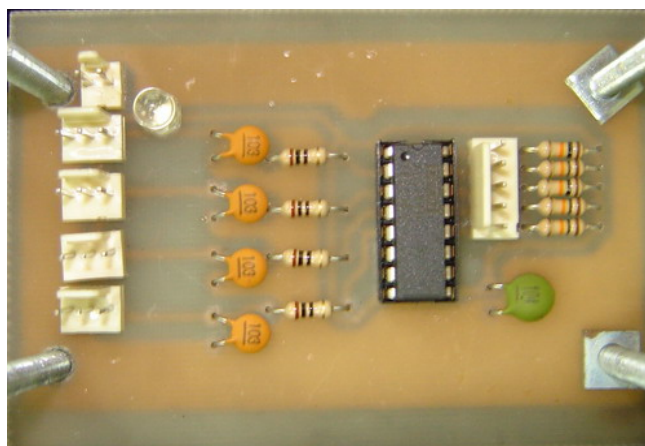


รูปที่ 3.4 (ก) Functional Diagram ของ ADS7841

(ข) Top View ของ ADS7841

3.1.3 การลงอุปกรณ์อิเล็กทรอนิกส์ตามที่ได้ออกแบบไว้

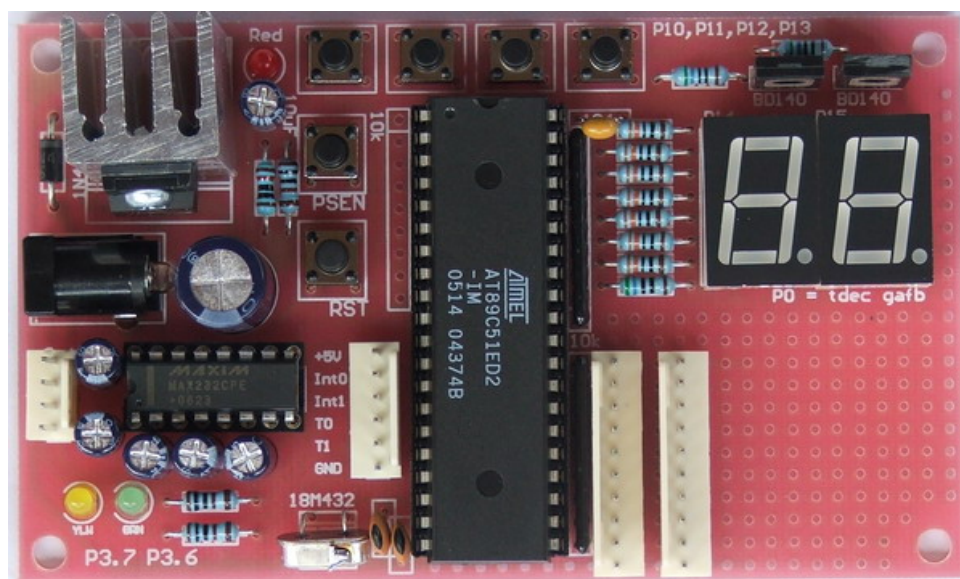
นำอุปกรณ์ที่ได้ออกแบบและจัดเตรียมมาประกอบลงบนแผ่นปริ้นท์ที่ได้ดังรูปที่ 3.5



รูปที่ 3.5 ชุดวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล

3.2 วงจรไมโครคอนโทรลเลอร์

ได้เลือกใช้วงจรไมโครคอนโทรลเลอร์สำเร็จรูป ในตระกูล MCS-51 ที่ใช้ไอซีเบอร์ AT89C51ED2 ของ ATMEL ดังที่ได้แสดงวงจรรวมไว้ดังรูปที่ 3.6

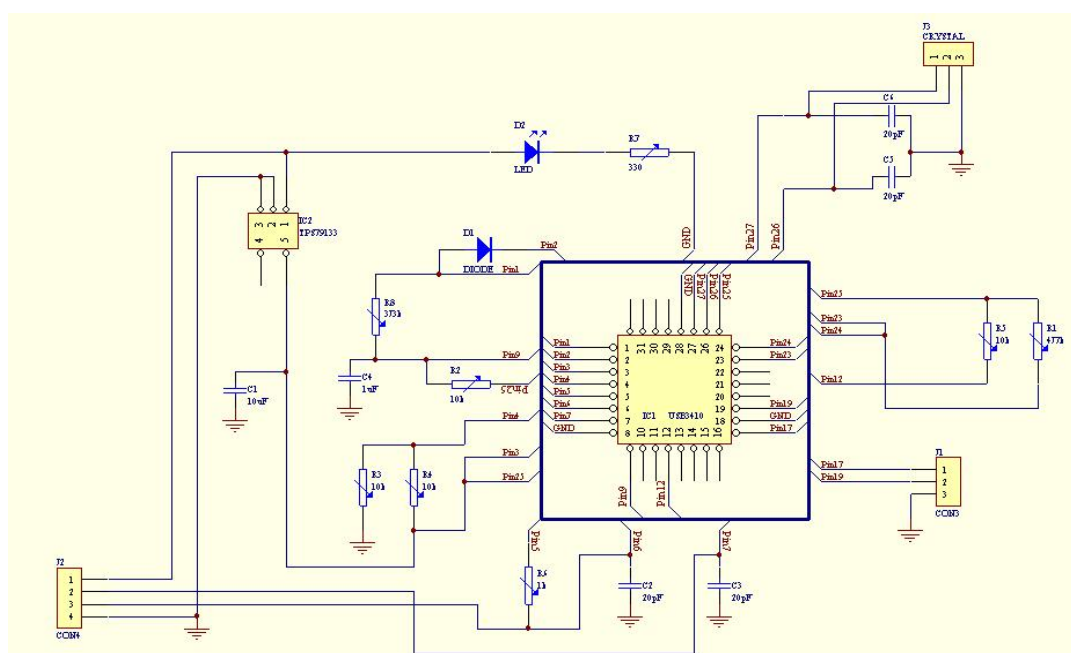


รูปที่ 3.6 ชุดวงจรไมโครคอนโทรลเลอร์ AT89C51ED2

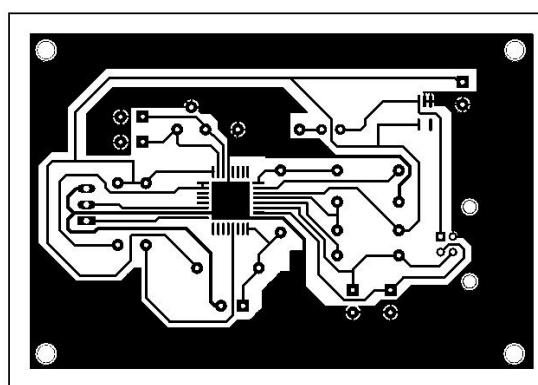
3.3 วงจรเชื่อมต่อ USB

3.3.1 การออกแบบลายวงจรเชื่อมต่อ USB ลงบนแผ่นปริ้นท์

จากรูปที่ 3.7 ได้แสดงถึงวงจรเชื่อมต่อ USB ที่ได้ทำการออกแบบไว้โดยเลือกใช้ไอซีที่มีคุณสมบัติตามต้องการ จากนั้นทำการออกแบบลายวงจรด้วยโปรแกรม Protel 99SE เพื่อให้ได้ลายวงจรเป็นแผ่นลายทองแดงที่จะสามารถนำมาใช้งานได้ตามรูปที่ 3.8



รูปที่ 3.7 วงจรเชื่อมต่อ USB

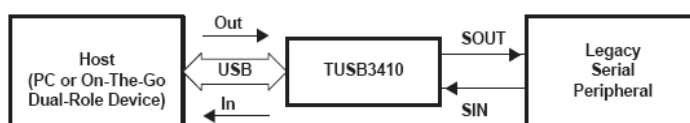


รูปที่ 3.8 ลายวงจรที่ได้จากโปรแกรม Protel99SE บนแผ่นปริ้นท์

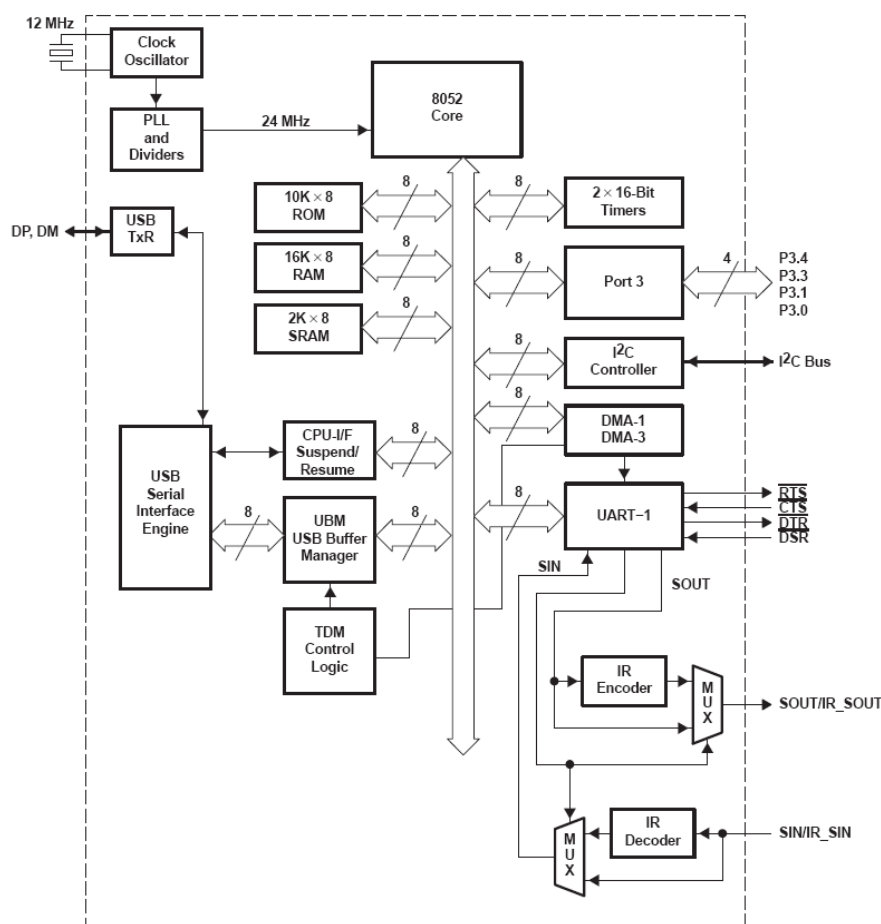
3.3.2 การเลือกใช้ไอซีในการออกแบบ

ในวงจรเชื่อมต่อ USB ได้เลือกใช้ไอซีจำนวน 2 ตัว คือ TUSB3410 และ TPS79133

TUSB3410 มีคุณสมบัติเป็นตัวเชื่อมต่อระหว่างอุปกรณ์ที่ส่งสัญญาณแบบอนุกรม ให้สามารถเชื่อมต่อกับคอมพิวเตอร์โดยผ่านทางพอร์ต USB ได้ และทำหน้าที่แทน พอร์ต RS-232 ได้ ซึ่งมีแผนผังการทำงานดังรูปที่ 3.9 (ก) และรูปที่ 3.9 (ข) และมีการจัดเรียงขาของไอซี ดังรูปที่ 3.10



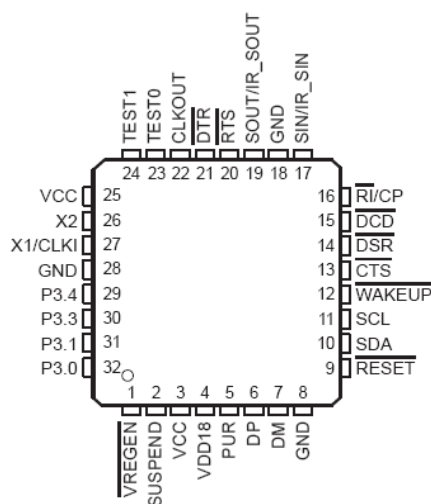
(ก)



(ข)

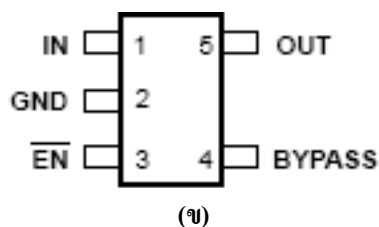
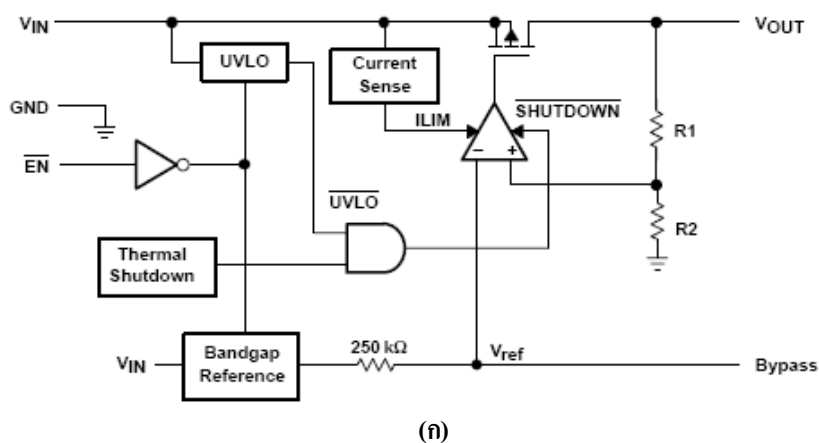
รูปที่ 3.9 (ก) Data Flow

(ข) USB-to-Serial (Single Channel) Controller Block Diagram



รูปที่ 3.10 Top View ของ TUSB3410

TPS79133 มีคุณสมบัติในการ แปลงระดับแรงดันที่ได้รับจากพอร์ต USB ของเครื่องคอมพิวเตอร์ จาก 5 โวลต์เป็น 3 โวลต์เพื่อจ่ายให้กับไอซี TUBS3410 ทำงาน โดยไอซีมีฟังก์การทำงานและการจัดเรียงขา ดังรูปที่ 3.11 (ก) และรูปที่ 3.11 (ข) ตามลำดับ



รูปที่ 3.11 (ก) Functional Block Diagram

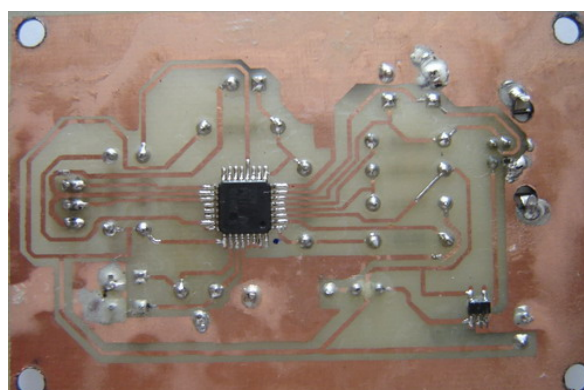
(ข) Top View ของ TPS79133

3.3.3 การลงอุปกรณ์อิเล็กทรอนิกส์ตามที่ได้ออกแบบ

นำอุปกรณ์ที่ได้ออกแบบและจัดเตรียมมาประกอบลงบนแผ่นปริ้นท์ได้ดังรูปที่ 3.12 (ก) และรูปที่ 3.12 (ข)



(ก)



(ข)

รูปที่ 3.12 (ก) ชุดวงจรเชื่อมต่อ USB ด้านบน

(ข) ชุดวงจรเชื่อมต่อ USB ด้านล่าง

3.4 สรุปผลการออกแบบ

การออกแบบวงจรทั้ง 3 ส่วนอันได้แก่ วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล, วงจรเชื่อมต่อ USB และวงจรไมโครคอนโทรลเลอร์ นั้นจะต้องใช้ความรู้และความเข้าใจในเรื่องของการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลที่ได้เรียนมาเป็นอย่างมาก และยังต้องใช้ความเข้าใจในเรื่องการเลือกใช้ไอซีเพื่อควบคุมการทำงานของวงจรเชื่อมต่อ USB ให้เกิดการติดต่อกันระหว่างไมโครคอนโทรลเลอร์กับเครื่องคอมพิวเตอร์

บทที่ 4

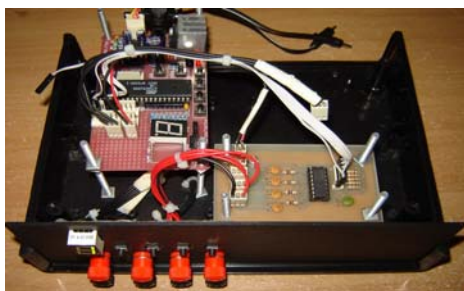
การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์สำหรับช่วยสอนปฏิบัติการ

4.1 การทดสอบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

เป็นการทดสอบเพื่อตรวจสอบความถูกต้องของข้อมูลที่ได้ออกมาจากวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลทางด้านเอาต์พุต โดยมีขั้นตอนดังต่อไปนี้

4.1.1 การติดตั้งชุดวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

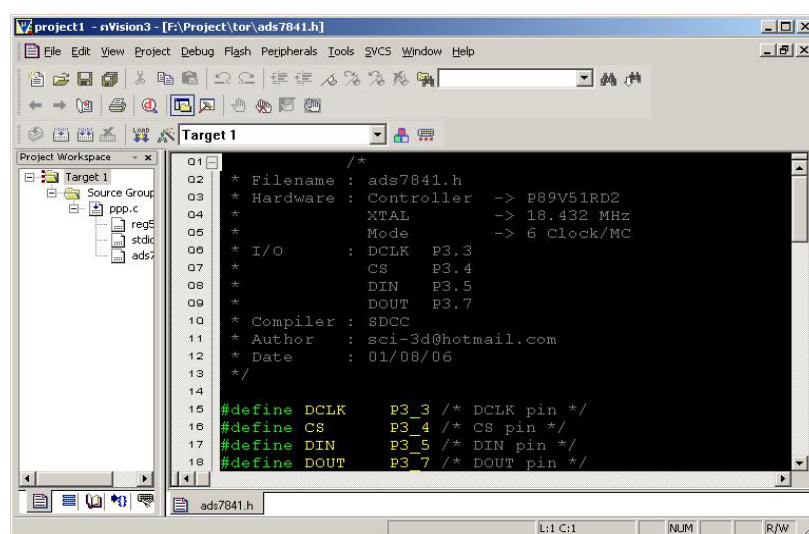
รูปที่ 4.1 แสดงถึงการจัดเรียงและติดตั้งอุปกรณ์เพื่อทำการทดสอบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล




รูปที่ 4.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลเพื่อการทดสอบ

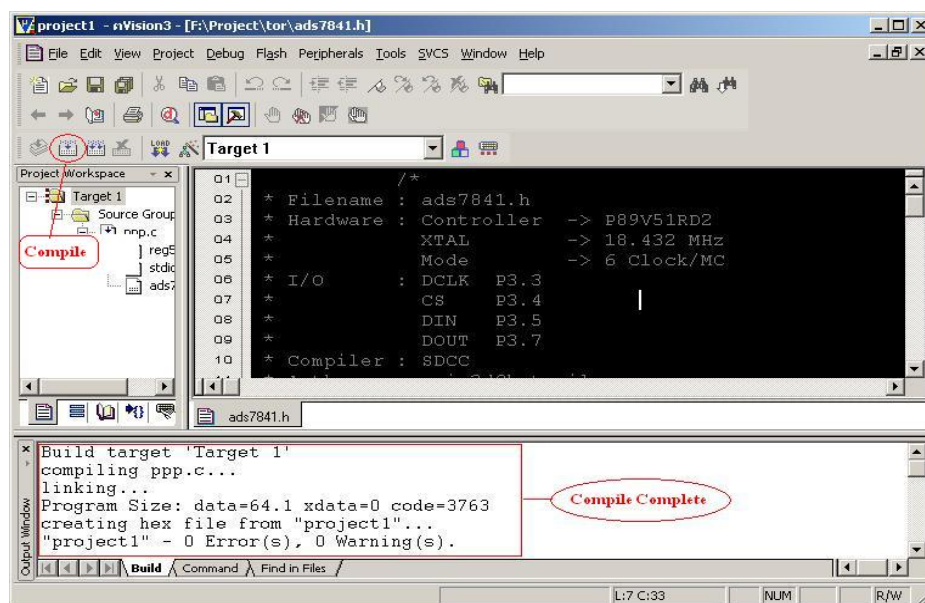
4.1.2 การเขียนโปรแกรมทดสอบ

เปิดโปรแกรม Keil70 ดังรูปที่ 4.2 เพื่อใช้ในการเขียนโปรแกรมทดสอบ



รูปที่ 4.2 แสดงโปรแกรม Keil70

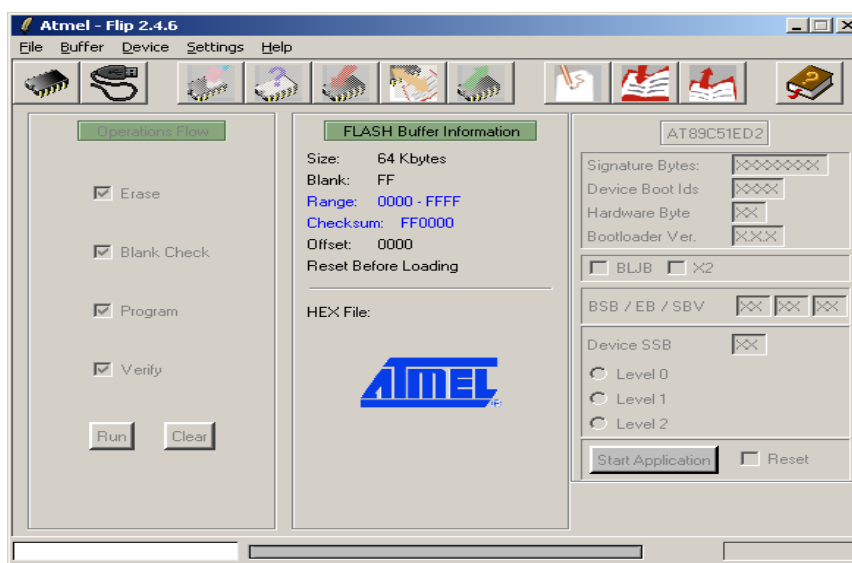
เมื่อทำการเขียนโปรแกรมเสร็จแล้วทำการคอมไพล์ (Compile) โดยกดปุ่ม  ถ้าสำเร็จ โดยไม่เกิดความผิดพลาดของโปรแกรมที่เขียนจะได้หน้าต่างดังรูปที่ 4.3



รูปที่ 4.3 แสดงการคอมไพล์สมบูรณ์

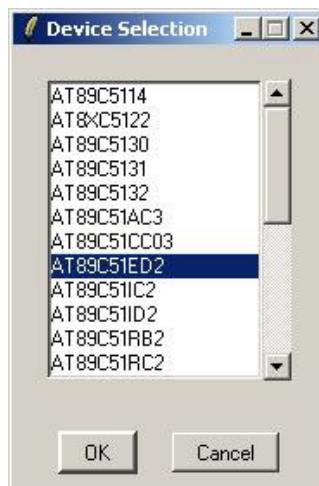
4.1.3 การโหลดโปรแกรมทดสอบ

เปิดโปรแกรม FLIP จะได้หน้าต่างดังรูปที่ 4.4 ใช้ในการโหลดโปรแกรมลงไมโครคอนโทรลเลอร์




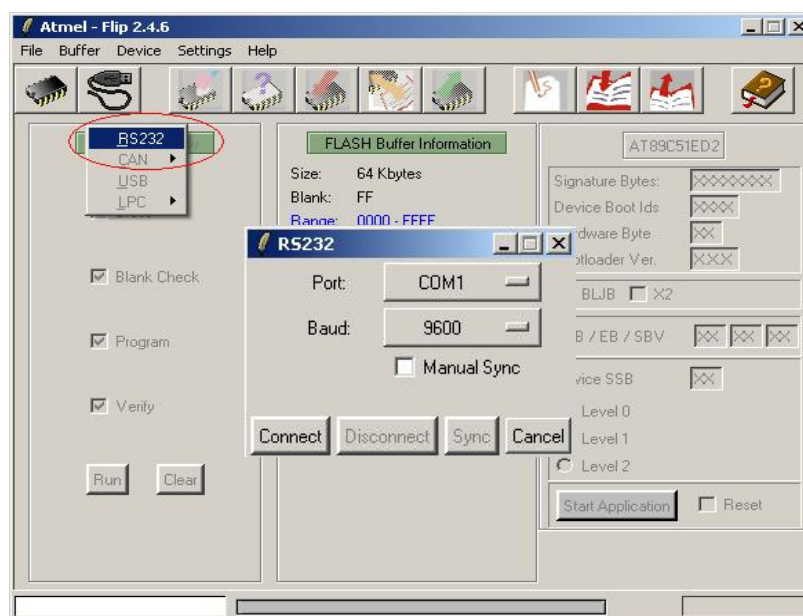
รูปที่ 4.4 หน้าต่างโปรแกรม FLIP

เลือกชนิดของไมโครคอนโทรลเลอร์ AT89C51ED2 ที่  จะได้หน้าต่างดังรูปที่ 4.5




รูปที่ 4.5 หน้าต่างเลือกชนิดไมโครคอนโทรลเลอร์

ตั้งค่าการเชื่อมต่อระหว่างโปรแกรมกับไมโครคอนโทรลเลอร์ที่  แล้วทำการเลือกเป็นแบบ RS-232, เลือกพอร์ต COM และตั้งค่าอัตราการส่งข้อมูล (Baud Rate) แล้วกด Connect ดังรูปที่ 4.6




รูปที่ 4.6 ตั้งค่าการเชื่อมต่อ

เมื่อตั้งค่าการเชื่อมต่อเรียบร้อยแล้วนั้น ทำการโหลดไฟล์ .HEX ที่ได้จากการคอมไพล์ ในโปรแกรม Keil70 โดยเลือกที่  จะปรากฏหน้าต่างดังรูปที่ 4.7

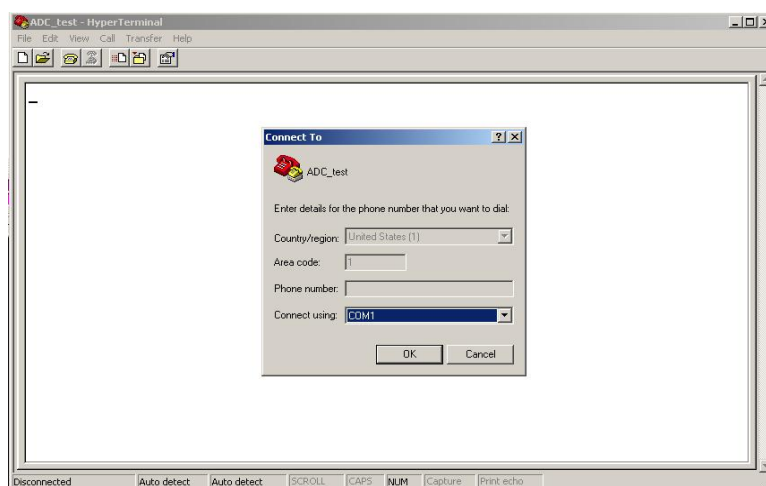


รูปที่ 4.7 โหลดไฟล์ .HEX

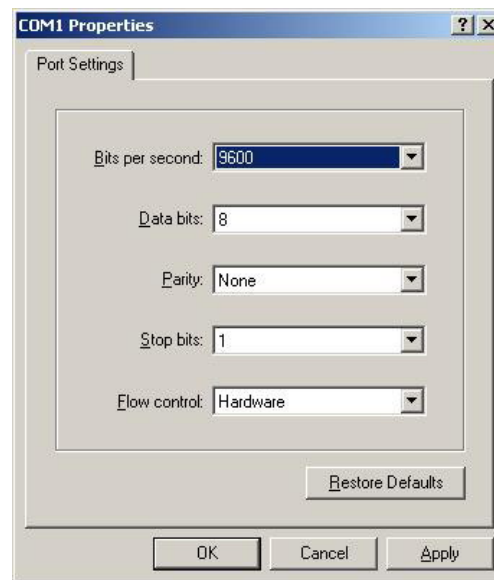
เมื่อเลือกไฟล์ .HEX ลงเรียบร้อยแล้วให้กดปุ่ม  โปรแกรมจะทำการ โหลดไฟล์ที่เลือกบอร์ดไมโครคอนโทรลเลอร์โดยอัตโนมัติ

4.1.4 ทดสอบวงจรแปลงสัญญาณแอนะล็อกไปเป็นสัญญาณดิจิทัล

เปิดโปรแกรม Hyper Terminal จะได้ดังรูปที่ 4.8 (ก) จากนั้นทำการตั้งค่าการเชื่อมต่อ โดยที่ ตั้งค่าอัตราการส่งข้อมูลไว้ที่ 9600 ดังรูปที่ 4.8 (ข)



(ก)

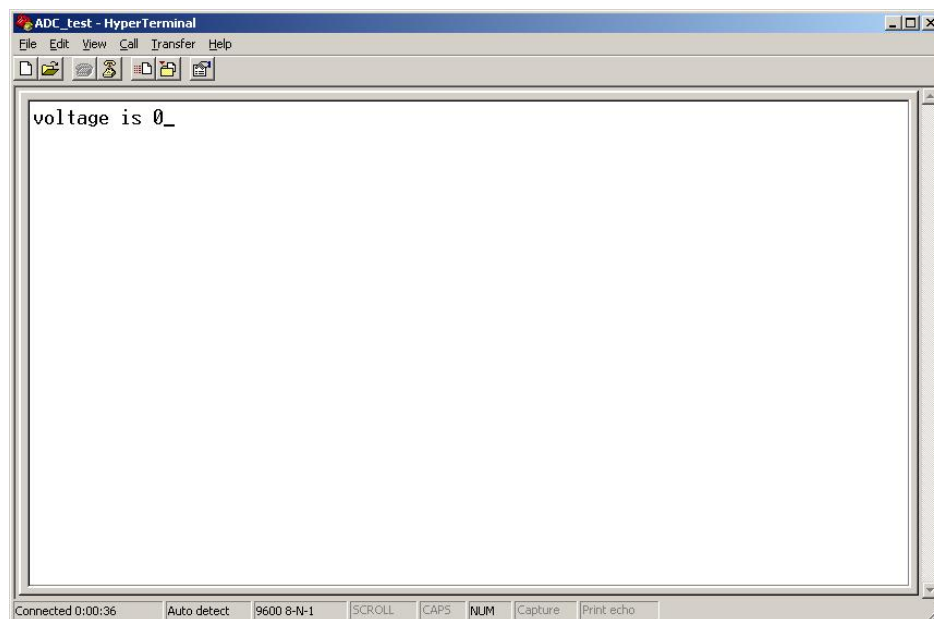


(ข)

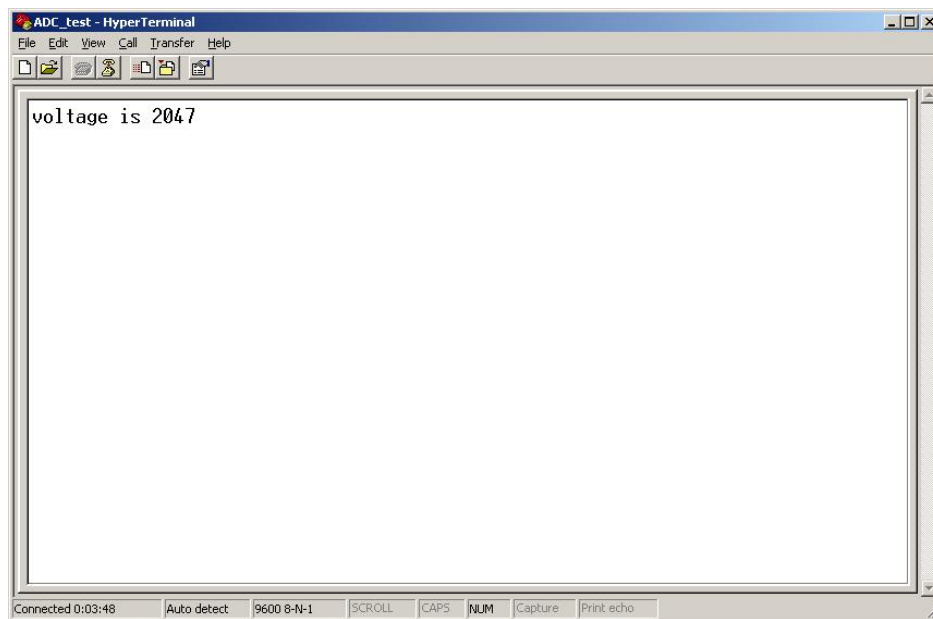
รูปที่ 4.8 (ก) หน้าต่าง Hyper Terminal

(ข) การตั้งค่าอัตราการส่งข้อมูล

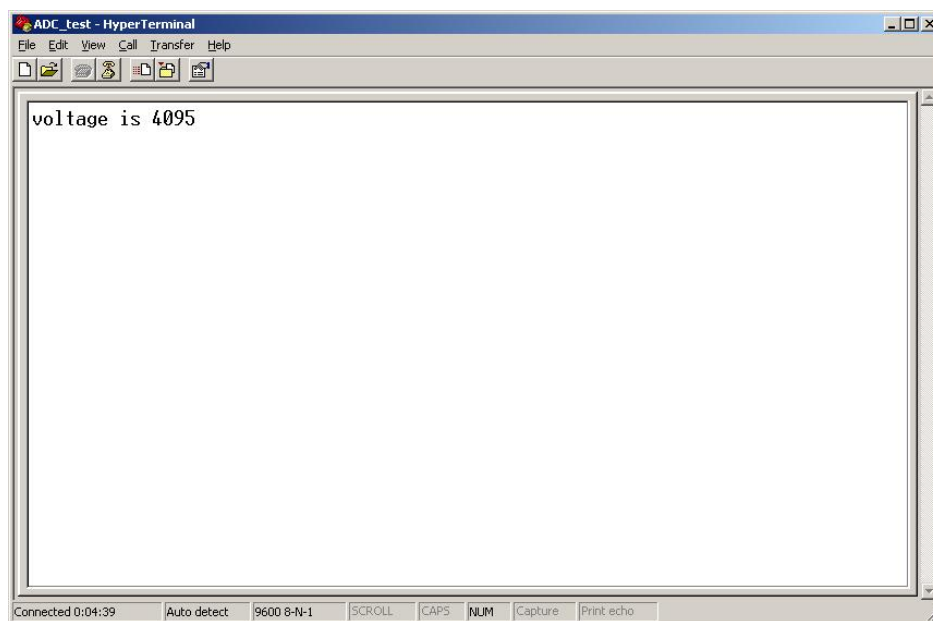
เมื่อทำการตั้งค่า Hyper Terminal เรียบร้อยแล้ว จึงทำการทดสอบวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลโดยการป้อนแรงดัน 0 โวลต์, 2.5 โวลต์ และ 5 โวลต์ ไปที่อนาลอกอินพุต แสดงได้ดังรูปที่ 4.9 (ก), รูปที่ 4.9 (ข) และ รูปที่ 4.9 (ค) ตามลำดับ ผลที่ได้คือ



(ก)



(ข)



(ค)

รูปที่ 4.9 (ก) ป้อนแรงดัน 0 โวลต์

(ข) ป้อนแรงดัน 2.5 โวลต์

(ค) ป้อนแรงดัน 5 โวลต์

จะเห็นว่า ค่าที่ออกมายังไม่ตรงกับค่าที่ต้องการให้แสดง โดยค่าที่ได้จากวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลตอนนี้เป็นค่าสเต็ป การที่จะทำให้ ค่าที่ได้ ออกมาเป็นค่าแรงดัน ทำได้โดยการหาค่าตัวคูณไปคูณค่าอนาลอกอินพุตที่เข้ามา โดยจะคิดที่แรงดันสูงสุดที่สามารถรับได้คือ 5 โวลต์

ขนาดแรงดันสูงสุดของสัญญาณอนาลอกอินพุต = 5 โวลต์

ขนาดของผลลัพธ์สูงสุดของการแปลงข้อมูล = 4095

ค่าแรงดันใน 1 สเต็ปของผลลัพธ์ = 5 โวลต์ / 4095

ดังนั้นเมื่อต้องการให้ค่าผลลัพธ์ที่อ่านได้จากวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลทางด้านเอาพุตที่เป็นค่าแรงดัน จะต้องนำค่าที่ส่งเข้าอนาลอกอินพุตไปคูณด้วย 5/4095 เพื่อให้ได้ค่าผลลัพธ์ที่แสดงเป็นค่าแรงดันที่ต้องการ

4.2 การทดสอบวงจรเชื่อมต่อ USB

เป็นการทดสอบเพื่อตรวจสอบว่าวงจรที่ได้ออกแบบขึ้นมานั้นสามารถที่จะเชื่อมต่อกับคอมพิวเตอร์ได้หรือไม่ และส่งข้อมูลไปยังคอมพิวเตอร์ ได้อย่างถูกต้องและแม่นยำ

4.2.1 การติดตั้งชุดวงจรเชื่อมต่อ USB เพื่อการทดสอบ

รูปที่ 4.10 แสดงถึงการจัดเรียงและต่ออุปกรณ์เพื่อทำการทดสอบวงจรเชื่อมต่อ USB

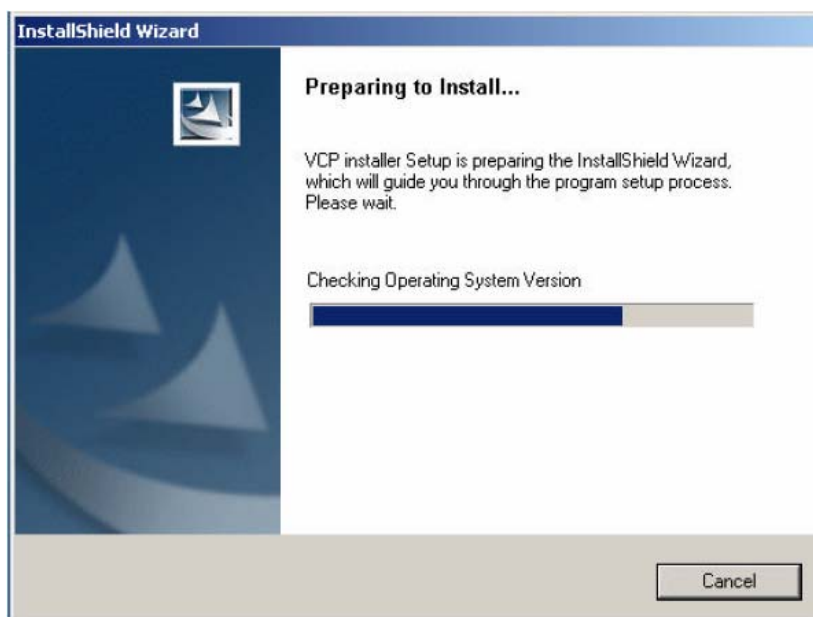


รูปที่ 4.10 การต่อวงจรเชื่อมต่อ USB เพื่อทดสอบ

4.2.2 การติดตั้งไดรเวอร์วงจรเชื่อมต่อ USB

การติดตั้งไดรเวอร์ของ TUSB3410 แบ่งเป็น 3 ขั้นตอนดังต่อไปนี้

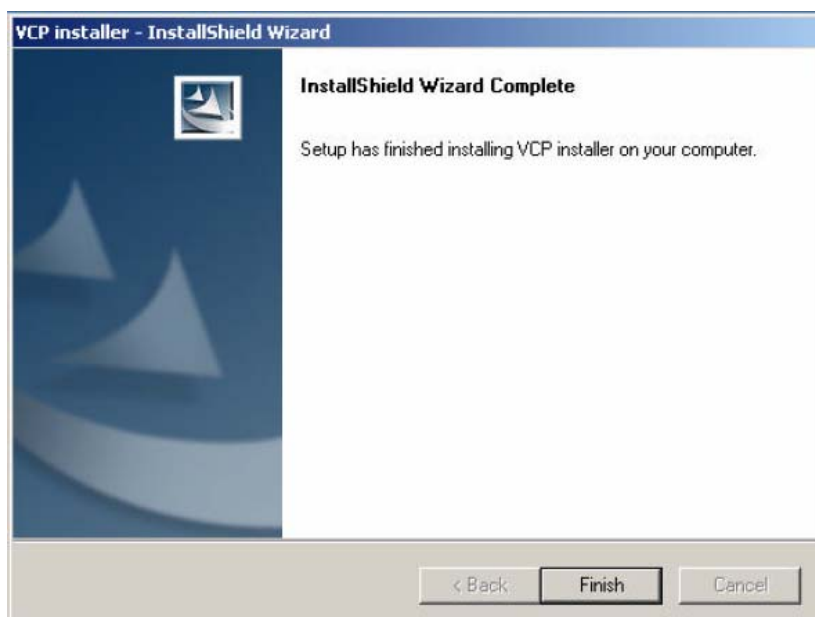
ขั้นตอนที่ 1 ทำการติดตั้งไดรเวอร์ด้วยไฟล์ setup.exe ซึ่งมีขั้นตอนดังรูปที่ 4.11(ก), รูปที่ 4.11 (ข) และรูปที่ 4.11 (ค) ตามลำดับ



(ก)



(ข)



(ก)

รูปที่ 4.11 (ก) Driver Installation Progress

(ข) License agreement

(ค) Pre-installation of drivers complete

ขั้นตอนที่ 2 ต่อวงจรเชื่อมต่อ USB เข้ากับเครื่องคอมพิวเตอร์ จะปรากฏหน้าต่างดังรูปที่ 4.12 (ก) ให้ติดตั้งฮาร์ดแวร์ (TUSB3410 device) เลือกที่ “Install the software automatically”



(ก)



(ข)

รูปที่ 4.12 (ก) Found New Hardware (“TUSB3410 Device”)

(ข) Device installation complete

ขั้นตอนที่ 3 ติดตั้งไดรเวอร์ของ “USB-Serial Port” ดังรูปที่ 4.13 (ก) เลือกที่ “Install the software automatically”



(ก)

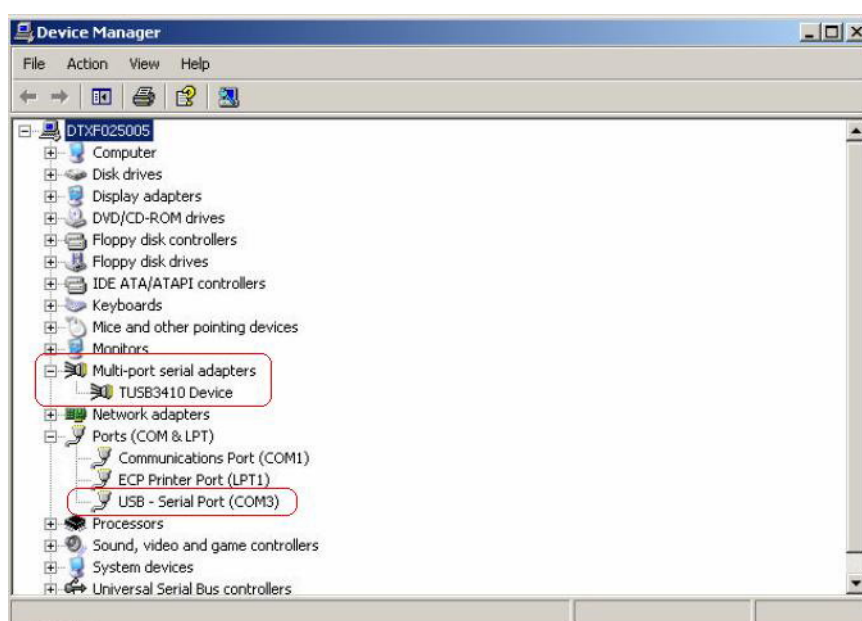


(๗)

รูปที่ 4.13 (ก) Found New Hardware (USB – Serial Port)

(๗) USB-Serial port installation complete

เมื่อทำการติดตั้งไดรเวอร์เสร็จเรียบร้อยแล้ว ทำการตรวจสอบดูว่าอุปกรณ์พร้อมใช้งานหรือไม่ที่ My Computer-> Properties -> Hardware -> Device Manager ดังรูปที่ 4.14

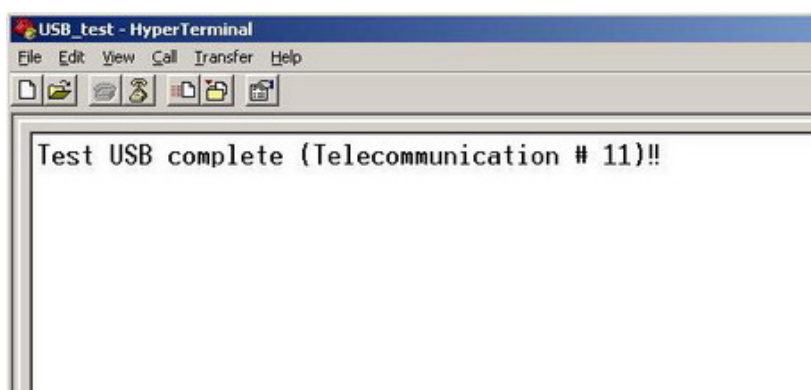


รูปที่ 4.14 Windows XP Device Manager

4.2.3 การทดสอบวงจรเชื่อมต่อ USB

เปิดโปรแกรม Hyper Terminal เลือก Connection using ให้ตรงกับพอร์ต USB ตั้งค่าอัตรา การส่งข้อมูลไว้ที่ 9600 ตามรูปที่ 4.8 (ข)

เมื่อเรียบร้อยแล้ว ทำการส่งข้อมูลจากวงจรเชื่อมต่อ USB โดยการเชื่อมต่อ Tx และ Rx ทางด้านอินพุตเข้าด้วยกันแล้วทดลองพิมพ์ ดูว่าข้อมูลที่ส่งไปขึ้นที่หน้าต่าง Hyper Terminal ตรงกับ ที่ส่งไปหรือไม่ ดังรูปที่ 4.15



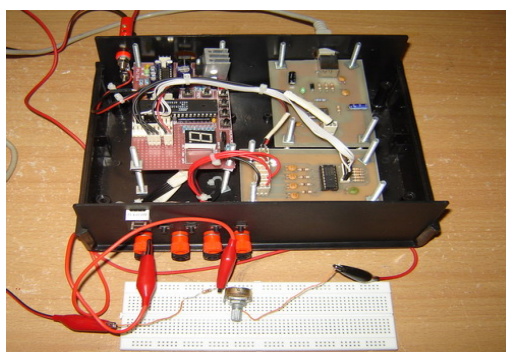
รูปที่ 4.15 ทดสอบส่งข้อมูลผ่านวงจรเชื่อมต่อ USB

4.3 การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

ในส่วนนี้จะเป็นการนำเอาวงจรทั้งหมดที่ได้ออกแบบและทำการสร้างขึ้น มาทำการ ทดสอบเพื่อตรวจสอบความถูกต้องของวงจรและ โปรแกรมที่ได้เขียนขึ้นมาเพื่อทำการประมวลผล และแสดงผลของข้อมูลที่ได้

4.3.1 การติดตั้งชุดอุปกรณ์วงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

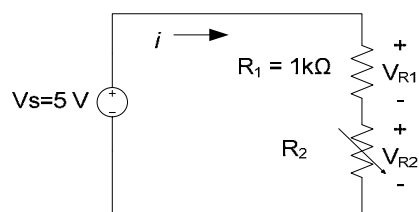
รูปที่ 4.16 แสดงถึงการจัดเรียงและต่ออุปกรณ์เพื่อทำการทดสอบวงจรเชื่อมต่อ คอมพิวเตอร์ช่วยสอนปฏิบัติการ



รูปที่ 4.16 วงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

4.3.2 การเขียนโปรแกรมควบคุมและแสดงผล

ทำการเขียนโปรแกรมควบคุมและโปรแกรมเพื่อสั่งการแสดงผลทางหน้าจอคอมพิวเตอร์ ด้วยโปรแกรม Keil70 โดยเปิดโปรแกรมขึ้นมาจะได้หน้าต่างตามรูปที่ 4.2 แล้วทำการเขียนโปรแกรมทดสอบโดยจะอ้างอิงจากทฤษฎีของวงจรแบ่งแรงดันมาเป็นแบบตัวอย่างวงจรทางไฟฟ้า โดยกำหนดแรงดันที่แหล่งจ่าย (V_s) เป็น 5 โวลต์ และความต้านทานตัวที่ 1 (R_1) = 1 กิโลโอห์ม แล้วเขียนโปรแกรมเพื่อหาค่า ความต้านทานตัวที่ 2 (R_2), กระแสของวงจร (I), แรงดันที่ตกคร่อมตัวต้านทานตัวที่ 1 (V_{R1}) และ 2 (V_{R2}) ตามลำดับ โดยใช้วงจรตามรูปที่ 4.17



รูปที่ 4.17 วงจรทดสอบ

จากรูปวงจรที่ 4.17 สามารถเขียนออกมาเป็นสมการได้ดังนี้

$$R_2 = \frac{I}{V_s - V_1} \quad (4.1)$$

$$I = \frac{V_s}{R_1 + R_2} \quad (4.2)$$

$$V_1 = IR_1 \quad (4.3)$$

$$V_2 = \frac{R_2}{R_1 + R_2} V_s \quad (4.4)$$

นำสมการที่ (4.1), (4.2), (4.3) และ (4.4) ไปประยุกต์ใช้เพื่อเขียนโปรแกรมแสดงผลผ่านทางหน้าจอคอมพิวเตอร์ เมื่อเขียนโปรแกรมเรียบร้อยแล้วจึงทำการคอมไพล์เพื่อตรวจสอบความผิดพลาดของโปรแกรมตามรูปที่ 4.3

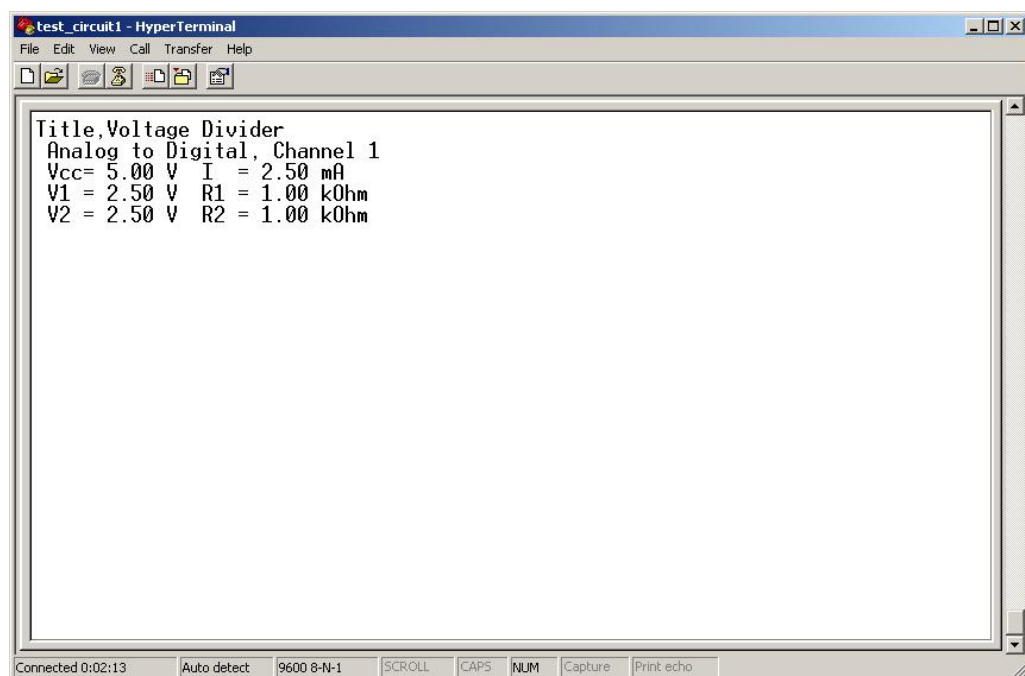
4.3.3 การโหลดโปรแกรมเพื่อทดสอบ

ทำการโหลดโปรแกรมที่ได้จากการเขียนโปรแกรมด้วย Keil70 ที่ผ่านการคอมไพล์แล้วเป็นไฟล์ .HEX ตามขั้นตอนที่ 4.1.3

4.3.4 การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการและโปรแกรม

โดยวงจรที่นำมาทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการนั้นเป็นวงจรแบ่งแรงดัน โดยที่เราจะทำการวัดค่าแรงดันที่ตกคร่อม R_2

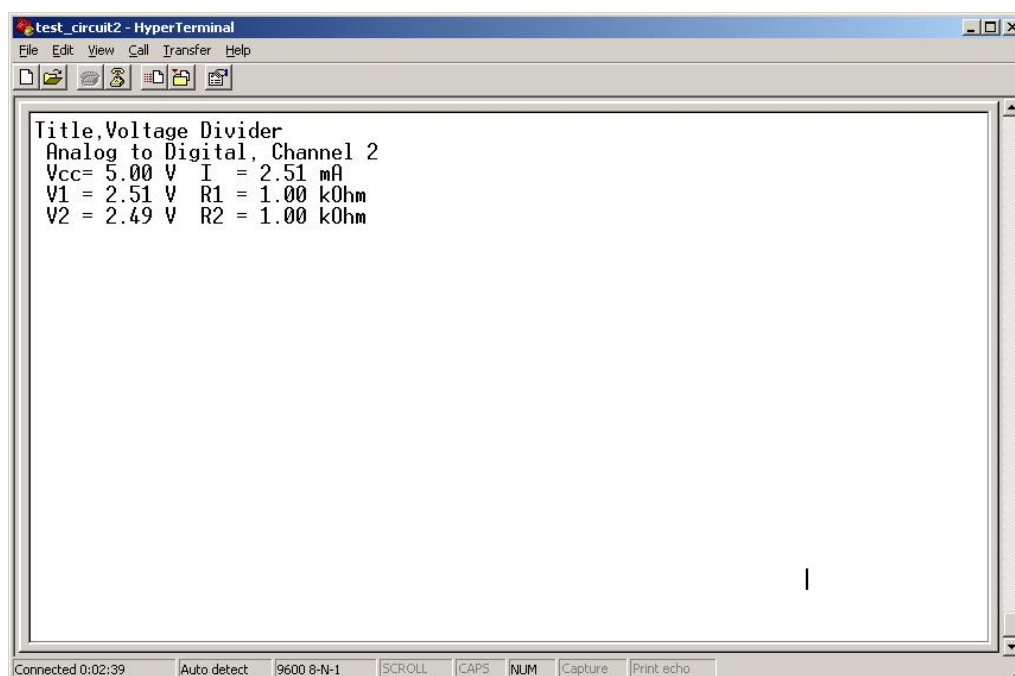
จะใช้โปรแกรม Hyper Terminal ในการแสดงผลโดยที่จะวัดค่าทั้งหมด 4 ช่องของอินพุตอนาลอก ได้ผลดังรูปที่ 4.18, รูปที่ 4.19, รูปที่ 4.20 และรูปที่ 4.21 พร้อมทั้งผลที่ได้จากการทดลองเปรียบเทียบกับผลที่ได้จากทฤษฎี ตามตารางที่ 4.1, ตารางที่ 4.2, ตารางที่ 4.3 และตารางที่ 4.4 ตามลำดับ ดังต่อไปนี้



รูปที่ 4.18 ผลการทดสอบอินพุตอนาลอกช่องที่ 1

ตารางที่ 4.1 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตนาลอกช่องที่ 1

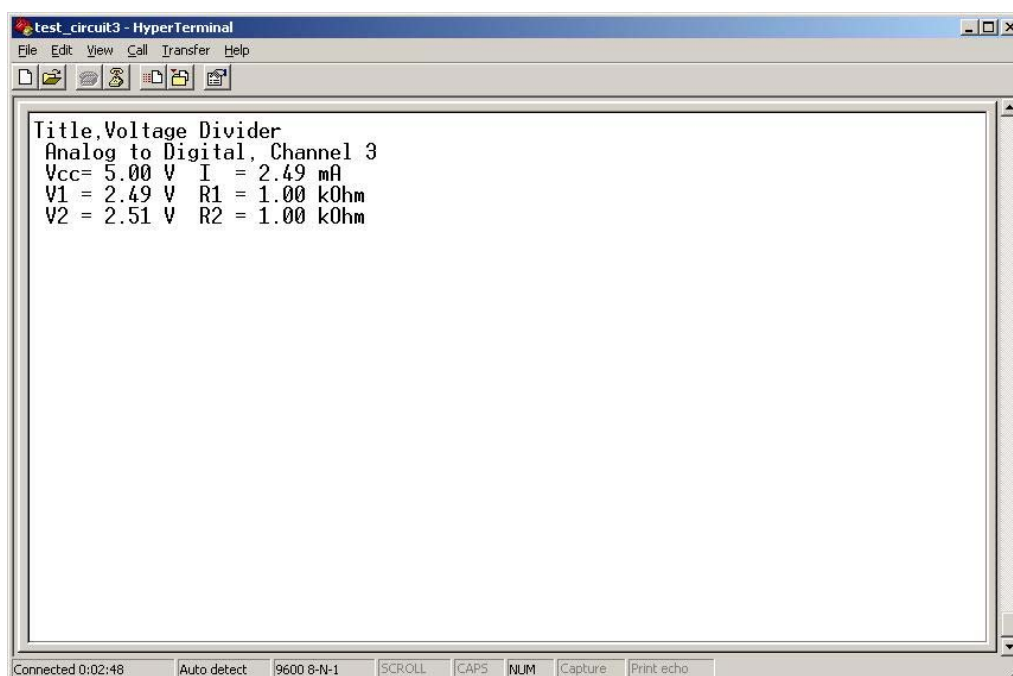
ผลที่ได้จากการทดลอง				ผลที่ได้จากการคำนวณ			
I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$	I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$
5	0	5	0	5	0	5	0
2.5	1	2.5	2.5	2.5	1	2.5	2.5
1.67	2	1.67	3.3	1.67	2	1.67	3.3
1.25	3	1.25	3.75	1.25	3	1.25	3.75
1	4	1	4	1	4	1	4
0.83	5	0.83	4.17	0.83	5	0.83	4.17
0.71	6	0.71	4.29	0.71	6	0.71	4.29
0.63	7	0.63	4.37	0.63	7	0.63	4.37
0.53	8	0.56	4.44	0.53	8	0.56	4.44
0.50	9	0.50	4.50	0.50	9	0.50	4.50
0.49	9.19	0.49	4.51	0.49	9.19	0.49	4.51



รูปที่ 4.19 ผลการทดสอบอินพุตนาลอกช่องที่ 2

ตารางที่ 4.2 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตนาลอกช่องที่ 2

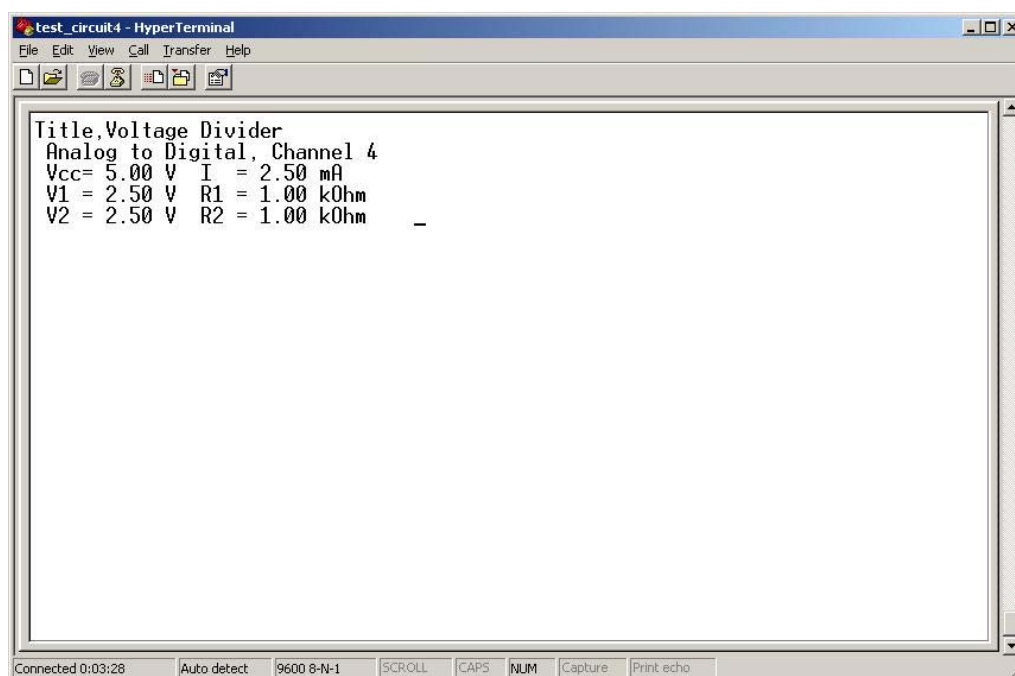
ผลที่ได้จากการทดลอง				ผลที่ได้จากการคำนวณ			
I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$	I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$
5	0	5	0	5	0	5	0
2.5	1	2.5	2.5	2.5	1	2.5	2.5
1.67	2	1.67	3.3	1.67	2	1.67	3.3
1.25	3	1.25	3.75	1.25	3	1.25	3.75
1	4	1	4	1	4	1	4
0.83	5	0.83	4.17	0.83	5	0.83	4.17
0.71	6	0.71	4.29	0.71	6	0.71	4.29
0.63	7	0.63	4.37	0.63	7	0.63	4.37
0.53	8	0.56	4.44	0.53	8	0.56	4.44
0.50	9	0.50	4.50	0.50	9	0.50	4.50
0.49	9.19	0.49	4.51	0.49	9.19	0.49	4.51



รูปที่ 4.20 ผลการทดสอบอินพุตนาลอกช่องที่ 3

ตารางที่ 4.3 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตนาลอกช่องที่ 3

ผลที่ได้จากการทดลอง				ผลที่ได้จากการคำนวณ			
I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$	I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$
5	0	5	0	5	0	5	0
2.5	1	2.5	2.5	2.5	1	2.5	2.5
1.67	2	1.67	3.3	1.67	2	1.67	3.3
1.25	3	1.25	3.75	1.25	3	1.25	3.75
1	4	1	4	1	4	1	4
0.83	5	0.83	4.17	0.83	5	0.83	4.17
0.71	6	0.71	4.29	0.71	6	0.71	4.29
0.63	7	0.63	4.37	0.63	7	0.63	4.37
0.53	8	0.56	4.44	0.53	8	0.56	4.44
0.50	9	0.50	4.50	0.50	9	0.50	4.50
0.49	9.21	0.49	4.51	0.49	9.21	0.49	4.51



รูปที่ 4.21 ผลการทดสอบอินพุตนาลอกช่องที่ 4

ตารางที่ 4.4 ผลการทดสอบวงจรตัวอย่างทางไฟฟ้า จากอินพุตอนาล็อกช่องที่ 4

ผลที่ได้จากการทดลอง				ผลที่ได้จากการคำนวณ			
I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$	I (mA)	$R_2(k\Omega)$	$V_{R1}(V)$	$V_{R2}(V)$
5	0	5	0	5	0	5	0
2.5	1	2.5	2.5	2.5	1	2.5	2.5
1.67	2	1.67	3.3	1.67	2	1.67	3.3
1.25	3	1.25	3.75	1.25	3	1.25	3.75
1	4	1	4	1	4	1	4
0.83	5	0.83	4.17	0.83	5	0.83	4.17
0.71	6	0.71	4.29	0.71	6	0.71	4.29
0.63	7	0.63	4.37	0.63	7	0.63	4.37
0.53	8	0.56	4.44	0.53	8	0.56	4.44
0.50	9	0.50	4.50	0.50	9	0.50	4.50
0.49	9.29	0.49	4.51	0.49	9.29	0.49	4.51

4.4 สรุปผลการทดสอบวงจร

4.4.1 การทดสอบวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล

เมื่อทำการทดสอบวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล ผลที่ได้ ถ้าต้องการให้เกิดการแสดงผลเป็นระดับแรงดันที่หน้าจอกอมพิวเตอร์นั้น จะต้องหาค่าตัวคูณเพื่อเปลี่ยนค่าที่ได้ทางด้านเอาต์พุตจากค่าสแต็ปไปเป็นค่าแรงดัน

4.4.2 การทดสอบวงจรเชื่อมต่อ USB

หลังจากที่ทำการทดสอบวงจรเชื่อมต่อ USB แล้วนั้นจะต้องใช้ความระมัดระวังในเรื่องของการใช้งาน ในบางครั้งอาจเกิดการเสียหายของตัวไอซีได้ง่าย

4.4.3 การทดสอบวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

จากผลการทดลองที่ได้จากทั้งหมด 4 ช่องสัญญาณอินพุตอนาล็อก ผลที่ได้จากการคำนวณและผลที่ได้จากการทดลองจริงนั้น มีค่าที่ไม่แตกต่างกันนั้นคือ การเขียน โปรแกรมและวงจรที่ใช้งาน ไม่เกิดข้อผิดพลาดใด ๆ

บทที่ 5

สรุปและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงการทำงานโดยรวมทั้งหมด โดยอธิบายส่วนประกอบของวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ และผลการทำงานโดยรวม รวมทั้งปัญหาที่เกิดขึ้นและข้อจำกัดในการทดลอง

5.1 ส่วนประกอบของวงจรเชื่อมต่อคอมพิวเตอร์ช่วยสอนปฏิบัติการ

- 5.1.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล
- 5.1.2 วงจรไมโครคอนโทรลเลอร์
- 5.1.3 วงจรเชื่อมต่อ USB

5.2 ปัญหาที่พบในขั้นตอนการทำงาน

- 5.2.1 เนื่องจากไอซีมีขนาดเล็กมาก อีกทั้งผู้ปฏิบัติการขาดความชำนาญในการบัดกรีอุปกรณ์จึงทำให้ไอซีบางตัว เกิดความเสียหายไม่สามารถทำงานได้
- 5.2.2 ไอซีส่วนใหญ่ที่นำมาใช้ในการออกแบบนั้นหาซื้อได้ยากและไม่มีขายในเมืองไทย จึงทำให้เสียเวลาในการสั่งซื้อจากต่างประเทศ
- 5.2.3 การออกแบบลายวงจรที่ได้ เส้นลายทองแดงมีขนาดที่ค่อนข้างเล็กเนื่องจากขาของไอซีที่มีขนาดเล็ก จึงเกิดการช็อตกันของลายวงจรได้ง่าย
- 5.2.4 ขาดความเข้าใจในคุณสมบัติ และการนำไปใช้งานของอุปกรณ์บางตัว

5.3 ข้อจำกัดของโครงการ

- 5.3.1 ไม่มีเครื่องมือที่ใช้ในการบัดกรีอุปกรณ์ขนาดเล็ก ทำให้ไอซีขนาดเล็กเสียหายบ่อยซึ่งเป็นการสิ้นเปลืองและเสียเวลา
- 5.3.2 ค่าแรงดันอินพุตที่วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลสามารถรับได้สูงสุดเพียง 5 โวลต์เท่านั้นจึงทำให้ไม่สามารถทำงานที่ค่าแรงดันสูงมากไม่ได้

5.4 ผลที่ได้จากโครงการ

- 5.4.1 ทำให้มีความรู้ความเข้าใจในหลักการทำงานของวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและสามารถนำไปประยุกต์ใช้ในอนาคตได้
- 5.4.2 ได้เรียนรู้วิธีการออกแบบลายวงจร และขั้นตอนการออกแบบผลิตภัณฑ์
- 5.4.3 มีความเข้าใจหลักการทำงาน และวิธีการออกแบบจากการศึกษาข้อมูลของไอซีแต่ละตัว

5.5 แนวทางการพัฒนาต่อ

- 5.5.1 ปรับปรุงวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลให้สามารถรับค่าแรงดันอินพุตที่มากกว่า 5 โวลต์
- 5.5.2 ปรับปรุงวงจรเชื่อมต่อ USB ให้เป็น Plug & Play โดยไม่ต้องติดตั้งไดรเวอร์ให้ยุ่งยากต่อการใช้งาน
- 5.5.3 ปรับปรุงและพัฒนาโปรแกรมที่ใช้แสดงผลให้มีความหลากหลายและสวยงาม

บรรณานุกรม

- [1] สันติ นุราชและคณะ. , “เรียนรู้ไมโครคอนโทรลเลอร์ MCS-51 ฉบับภาษา C”, Micro Research Technology. Ltd, Part, ปทุมธานี.
- [2] <http://www.pantip.com/tech/newscols/column/15-07-00/usb1/usb.html>(USB)
<http://www.pantip.com/tech/newscols/column/01-08-00/usb2/usb2.html>(USB2)
<http://www.justusers.net/knownledges/usb.htm>
- [3] สราวุฒิ สุจิตรจร และกิตติ อัครกกิจมงคล (แปลและเรียบเรียง). “วงจรไฟฟ้า Electric Circuit.” พิมพ์ครั้งที่ 1, เพียร์สัน เอ็ดดูเคชั่น อินโดไชน่า, กรุงเทพฯ: 2547
- [4] บัณฑิต จามรภูติ. , “คู่มือการใช้งาน Protel 99.”, บัณฑิตเพรส, เชียงใหม่: 2544

ภาคผนวก (ก)

โปรแกรมควบคุมและสั่งการให้แสดงผลผ่านทางหน้าจอคอมพิวเตอร์ ของ “วงจรแบ่งแรงดัน”

```
#include <REG51xD2.H>
#include<stdio.h>
#include <ads7841.h>
unsigned char data_show[]={0x88,0xEE,0x92,0xA2,0xE4};
sbit sw1 = P1^0;
sbit sw2 = P1^1;
sbit sw3 = P1^2;
sbit sw4 = P1^3;
void SerialInt(void);
void delay(unsigned int);

main()
{  unsigned char ch;
    float a;
    float b;
    float R2;
    float I;
    float c;
    SerialInt();
    P1=0xDF;
    ch=1;
    while(1)
    {
        if (sw1 == 0 ){ch=1;}
        if (sw2 == 0 ){ch=2;}
        if (sw3 == 0 ){ch=3;}
        if (sw4 == 0 ){ch=4;}
```

```

a = analog(ch-1);
b=a*5/4095;
R2=b*1/(5-b);
I=b/R2;
c=5-b;

if (ch == 1 ){P0=data_show[ch]; putchar(12);
printf("\rTitle,Voltage Divider \n");
printf(" Analog to Digital, Channel 1\n");
printf(" Vcc= 5.00 V I = %.2f mA\n",I);
printf(" V1 = %.2f V R1 = 1.00 kOhm\n",c);
printf(" V2 = %.2f V R2 = %.2f kOhm  ",b,R2);
delay(60000);

        }

if (ch == 2 ){P0=data_show[ch]; putchar(12);
printf("\rTitle,Voltage Divider \n");
printf(" Analog to Digital, Channel 2\n");
printf(" Vcc= 5.00 V I = %.2f mA\n",I);
printf(" V1 = %.2f V R1 = 1.00 kOhm\n",c);
printf(" V2 = %.2f V R2 = %.2f kOhm  ",b,R2);
delay(60000);

        }

if (ch == 3 ){P0=data_show[ch]; putchar(12);
printf("\rTitle,Voltage Divider \n");
printf(" Analog to Digital, Channel 3\n");
printf(" Vcc= 5.00 V I = %.2f mA\n",I);
printf(" V1 = %.2f V R1 = 1.00 kOhm\n",c);
printf(" V2 = %.2f V R2 = %.2f kOhm  ",b,R2);
delay(60000);

        }

```

```

        if (ch == 4 ) {P0=data_show[ch]; putchar(12);

        printf("\rTitle,Voltage Divider \n");

        printf(" Analog to Digital, Channel 4\n");

        printf(" Vcc= 5.00 V I = %.2f mA\n",I);

        printf(" V1 = %.2f V R1 = 1.00 kOhm\n",c);

        printf(" V2 = %.2f V R2 = %.2f kOhm  ",b,R2);

        delay(60000);

        }

    }

}

void SerialInt(void)
{
    TR1=0;

    SCON = 0x52;  /* SCON0 */ /* setup serial port control */

    TMOD = 0x21;

    TCON = 0x69;  /* TCON */

    TH1 = 0xFB;  // 28800 at Clock 32 Mhz

    TR1=1;

}

void delay(unsigned int a)
{
    unsigned int i,b;

    for(i=0;i<=a;i++)

        {
            b=10000;

            while(b<=0)

                b--;

        }

    return;

}

```

โปรแกรมควบคุมการทำงานของวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

```
* Filename : ads7841.h
* Hardware : Controller -> P89V51RD2
*     XTAL     -> 18.432 MHz
*     Mode     -> 6 Clock/MC
*     I/O      : DCLK P3.3
*     CS       P3.4
*     DIN      P3.5
*     DOUT     P3.7
```

.....

```
#define DCLK  P3_3 /* DCLK pin */
#define CS      P3_4 /* CS pin */
#define DIN     P3_5 /* DIN pin */
#define DOUT   P3_7 /* DOUT pin */
```

```
int analog(char ch)
```

```
{
    unsigned char i, crt_byte;      /* Variable for counter */
    unsigned int dat;               /* Variable for storing analog value */
    switch(ch)                      /* Check selected channel for reading */
    {
        case 0 : crt_byte = 0x97; /* Select analog channel 0 */
                break;
        case 1 : crt_byte = 0xD7; /* Select analog channel 1 */
                break;
        case 2 : crt_byte = 0xA7; /* Select analog channel 2 */
                break;
        case 3 : crt_byte = 0xE7; /* Select analog channel 3 */
                break;
    }
}
```



```

        default: crt_byte = 0x97; /* Select analog channel 0 */
                break;
    }

    dat = 0;
    DCLK = 0;      /* Clear DCLK pin */
    DIN = 0;       /* Clear DIN pin */
    CS = 0;        /* Clear CS pin */

    for(i=0; i<8; i++) /* Loop to send control byte for ADS7841 */
    {
        DIN = crt_byte & 0x80; /* Send data bit to ADS7841 */
        DCLK = 0;              /* Clear clock pin */
        DCLK = 1;              /* Set clock pin */
        crt_byte = crt_byte<<1; /* Shift bit once next time */
    }

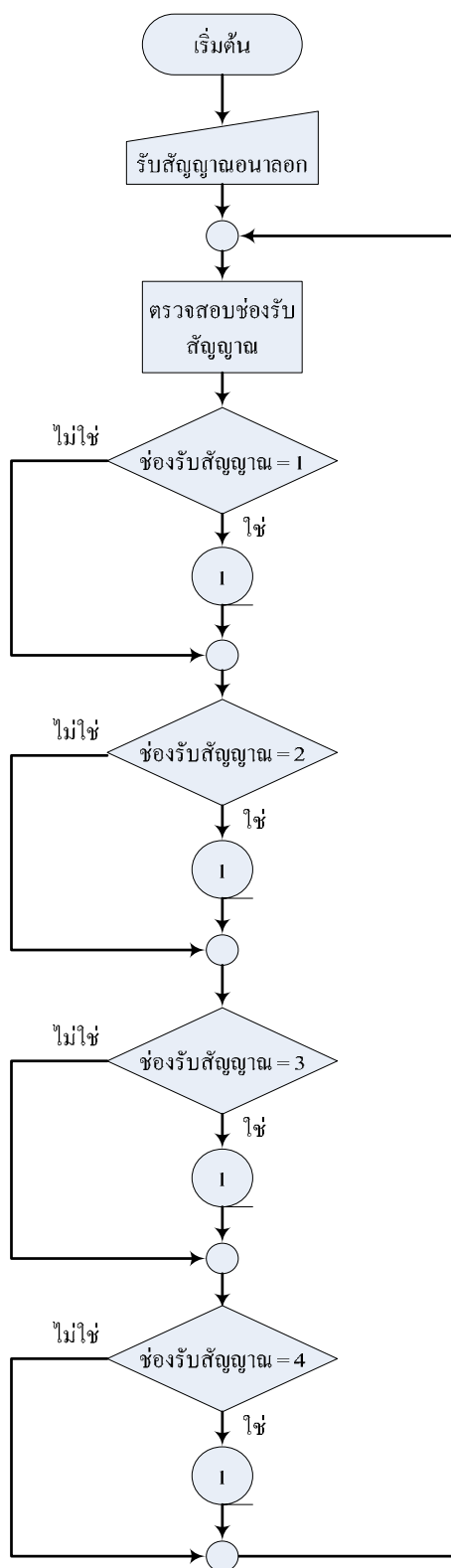
    DCLK = 0; /* Clear clock pin */
    DCLK = 1; /* Set clock pin */

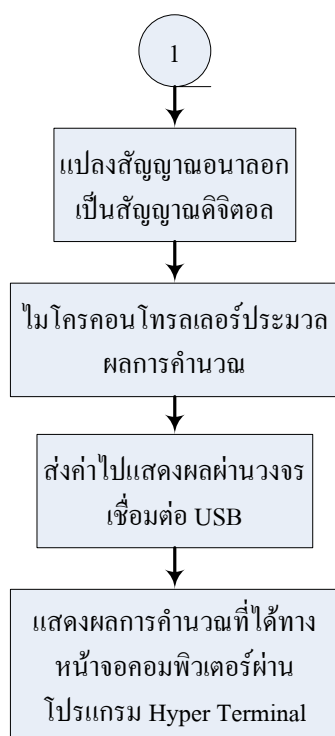
    for(i=0; i<12; i++) /* Loop to read 12-bit digital value from ADS7841 */
    {
        DCLK = 0; /* Clear clock pin */
        DCLK = 1; /* Set clock pin */
        dat = dat<<1; /* Shift bit once to store data next bit */
        dat = dat | DOUT; /* Read data bit from DOUT pin */
    }

```

```
for(i=0;i<4;i++)          /* Generate rising edge clock for success frame */
{
    DCLK = 0;              /* Clear clock pin */
    DCLK = 1;              /* Set clock pin */
}
    CS = 1;                /* Set CS pin */
return dat;
}
```

แผนผังการทำงานของโปรแกรมควบคุม



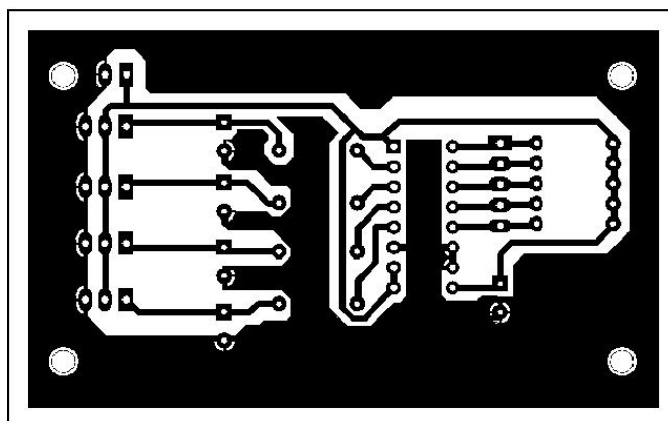


ภาคผนวก (ข)

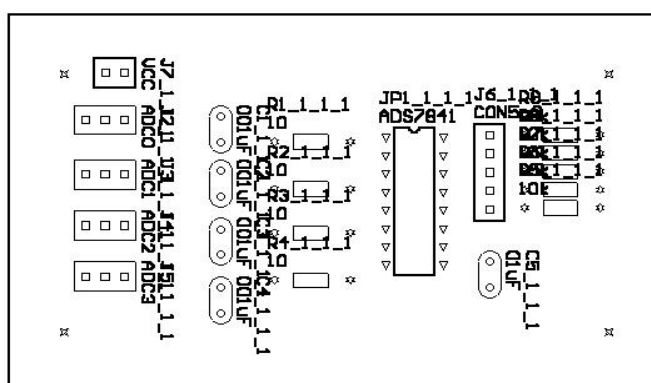
อุปกรณ์ที่ใช้ และการลงอุปกรณ์บนแผ่นปริ้น

วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

1. ตัวเก็บประจุ 0.01 μF 4 ตัว
2. ตัวเก็บประจุ 0.1 μF 1 ตัว
3. ตัวต้านทาน 100 Ω 4 ตัว
4. ตัวต้านทาน 10 $\text{k}\Omega$ 5 ตัว
5. ไอซีเบอร์ ADS7841 1 ตัว
6. LED 1 ตัว



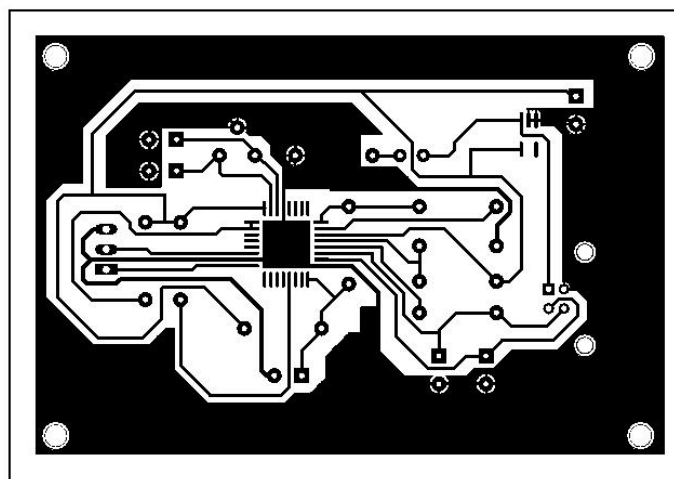
รูปลายวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล



รูปตำแหน่งของอุปกรณ์วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล

วงจรเชื่อมต่อ USB

1. ตัวเก็บประจุ	20	pF	4	ตัว
2. ตัวเก็บประจุ	1	μ F	1	ตัว
3. ตัวเก็บประจุ	10	μ F	1	ตัว
4. ตัวต้านทาน	10	k Ω	4	ตัว
5. ตัวต้านทาน	330	Ω	1	ตัว
6. ตัวต้านทาน	1	k Ω	1	ตัว
7. ตัวต้านทาน	4.7	k Ω	1	ตัว
8. ตัวต้านทาน	33	k Ω	1	ตัว
9. XTAL	12	MHz	1	ตัว
10. Diode	N4002		1	ตัว
11. ไอซีเบอร์	TUSB3410		1	ตัว
12. ไอซีเบอร์	TPS79133		1	ตัว
13. USB Connector (Type B)			1	ตัว
14. LED			1	ตัว



รูปลายวงจรของวงจรเชื่อมต่อ USB

ประวัติผู้จัดทำ



นายวิรพงษ์ ห้าวหาญ เกิดเมื่อวันที่ 6 มีนาคม พ.ศ. 2527 ภูมิลำเนาเดิมอยู่ บ้านเลขที่ 22/6 หมู่ที่ 3 ตำบลเพ อำเภอเมือง จัหวัระยอง สำเร็จการศึกษา ระดับชั้นมัธยมศึกษาตอนปลายจากโรงเรียนพรักษ์มาตาวิทยา อำเภอเมือง จังหวัดระยอง ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 สาขาวิชาวิศวกรรม โตรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นายอัฐพล เรืองบุตร เกิดเมื่อวันที่ 2 กรกฎาคม พ.ศ. 2528 ภูมิลำเนาเดิมอยู่ บ้านเลขที่ 36/5 หมู่ที่ 6 ตำบลบางพระ อำเภอศรีราชา จังหวัดชลบุรี สำเร็จ การศึกษาระดับชั้นมัธยมศึกษาตอนปลายจากโรงเรียนชลราษฎรอำรุง อำเภอเมือง จังหวัดชลบุรี ปีการศึกษา 2545 ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 สาขาวิชาวิศวกรรมโทรคมนาคม สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัย เทคโนโลยีสุรนารี จังหวัดนครราชสีมา